Software yang diperlukan:

1. Web Browser
2. XAMPP (dengan php versi 8.1.*)
3. Visual Studio Code
4. Composer
5. Git

File yang diperlukan :

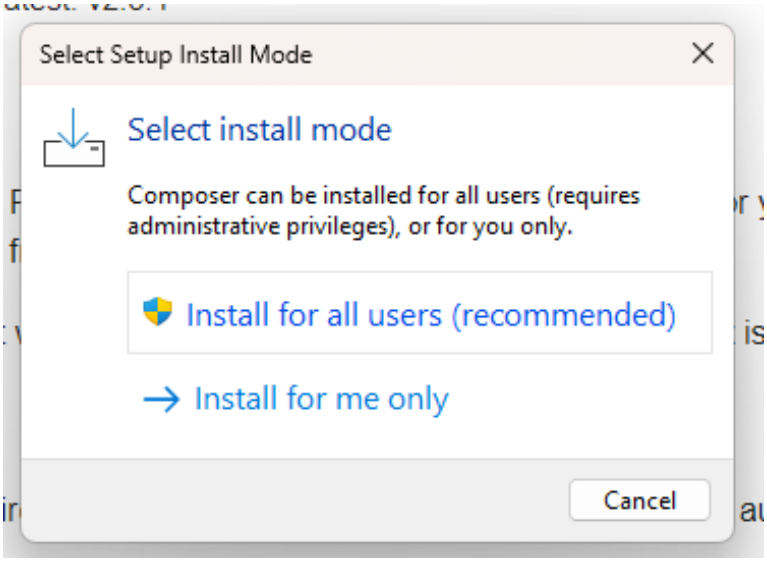
1. Folder proyek Laravel yang bisa di download atau di clone (https://github.com/MitsaFata/tanaman.git)

Langkah – Langkah:

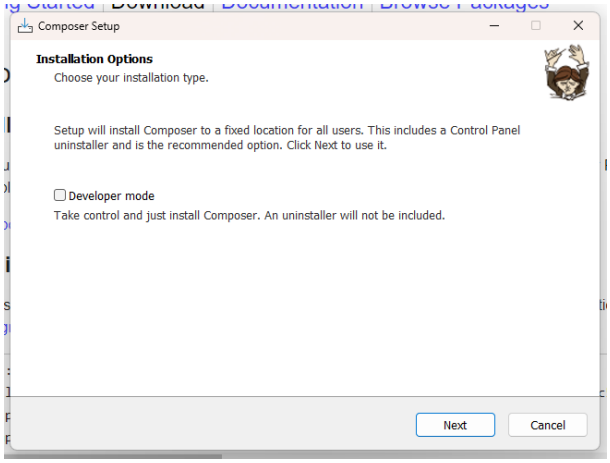
1. Install semua software yang diperlukan diatas. Karena Composer memerlukan cara tertentu, kami akan menjelaskan stepnya dibawah ini

   Cara install Composer

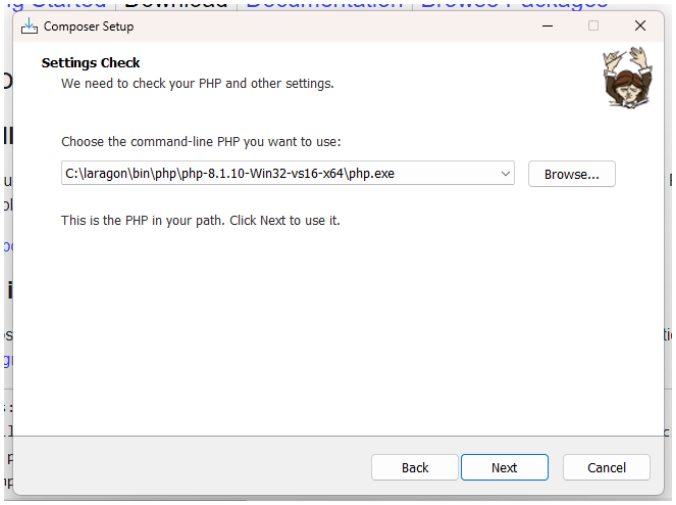
   1. Download file instalasi dari https://getcomposer.org/download/
   2. Jalankan file installasi tersebut.
   3. Akan muncul pop-up seperti berikut. Pilih install for all user untuk menginstall composer pada semua user, atau install for me only jika ingin menginstall pada user yang saat ini digunakan saja.

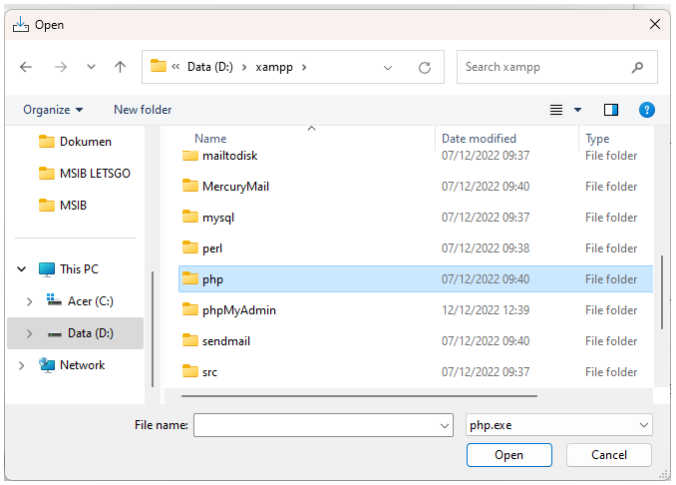


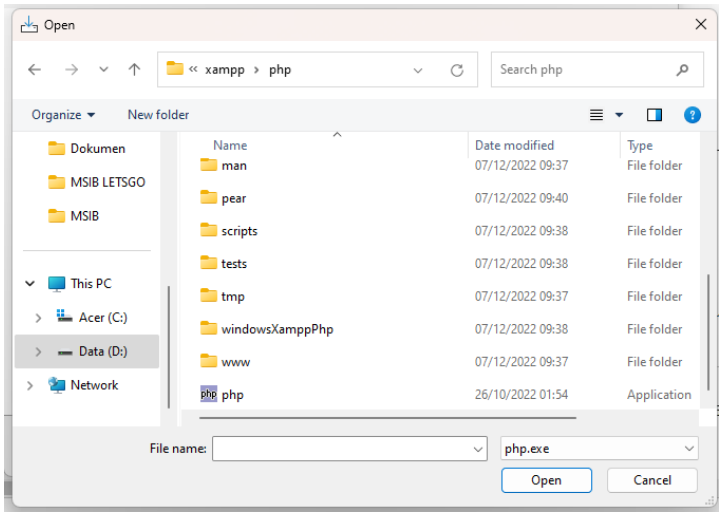
4. Kemudian, akan muncul pilihan developer. Bagian ini dilewati saja dengan mengklik next.

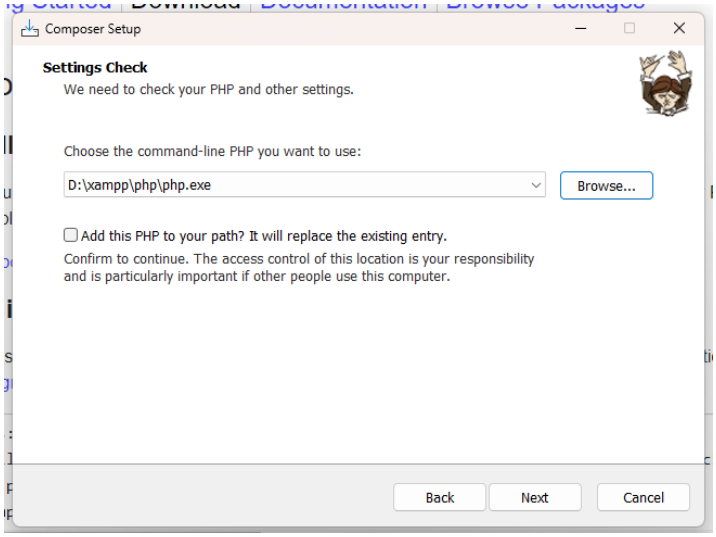


5. Lalu akan diminta memilih file php yang akan digunakan. Karena kita sudah menginstall XAMPP maka pilihlah saja file php.exe yang ada didalamnya. Lokasinya ada pada folder php dalam folder instalasi xampp.



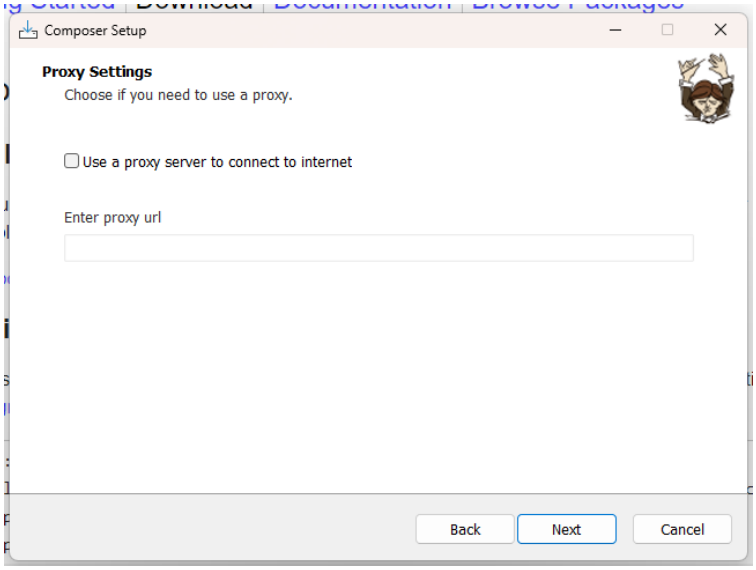




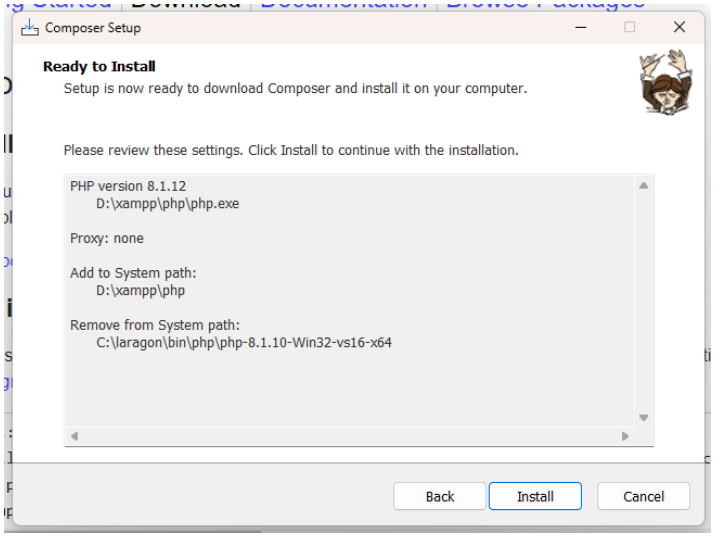


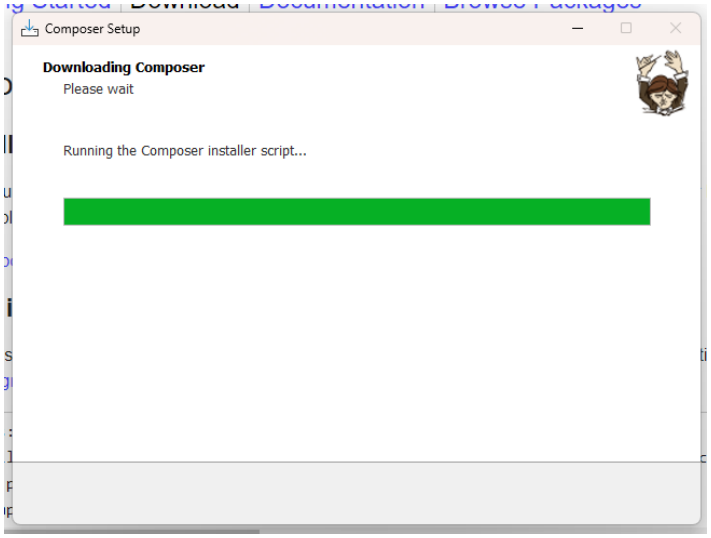
Lalu klik next.

6. Kemudian akan diberikan opsi menambahkan proxy. Kita akan melewati proses ini.

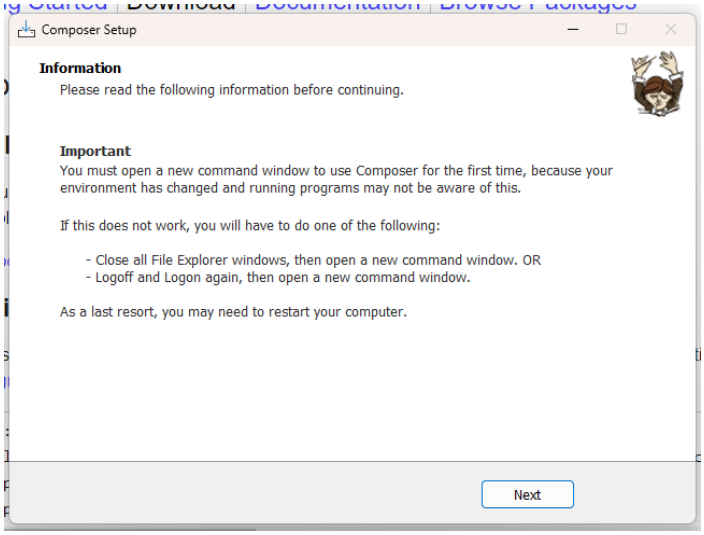


7. Kemudian pada bagian ini klik install dan tunggu hingga selesai

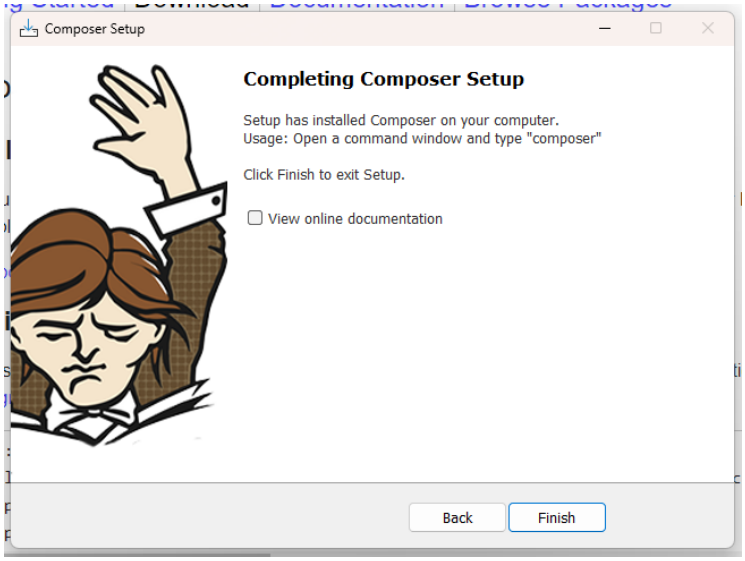




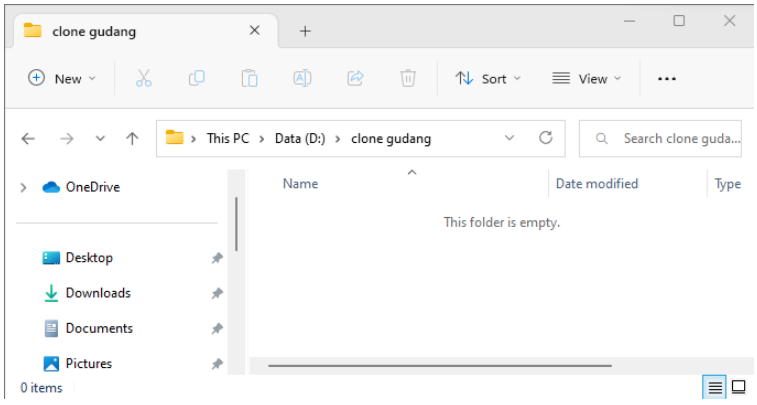
8. Setelah selesai, akan ditampilkan informasi untuk menjalankan composer pada command prompt untuk pertama kalinya.



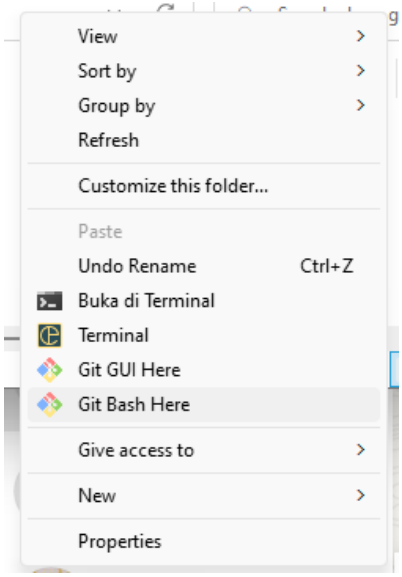
9. Terakhir klik finish



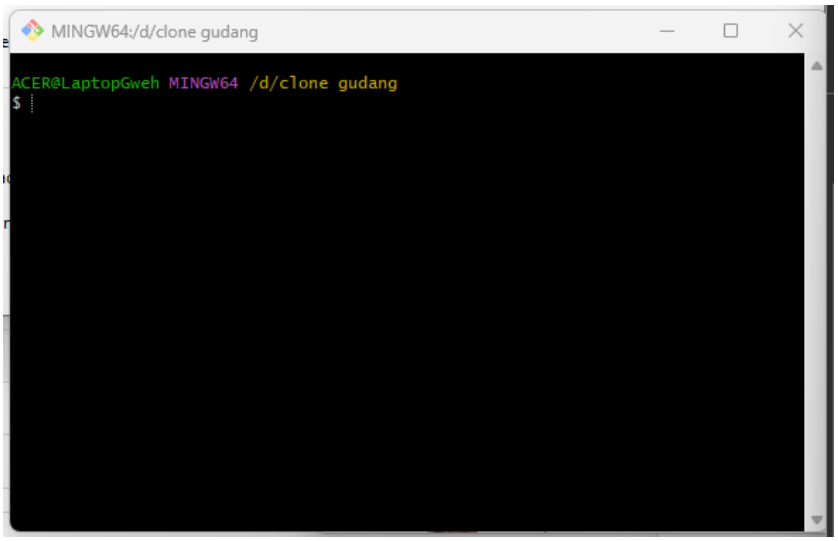
2. Kedua, kita harus melakukan clone repositori dari github. Pertama kita harus menentukan folder untuk menyimpan repositori. Disini kami menggunakan folder kosong Bernama clone Gudang.



Lalu klik kanan pada bagian yang kosong. Dan klik “git bash here”.



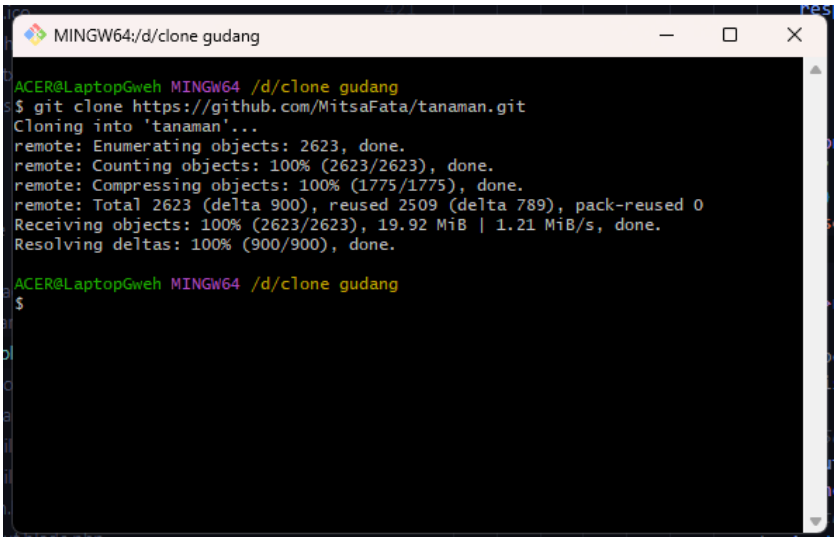
Setelahnya akan muncul tampilan seperti ini



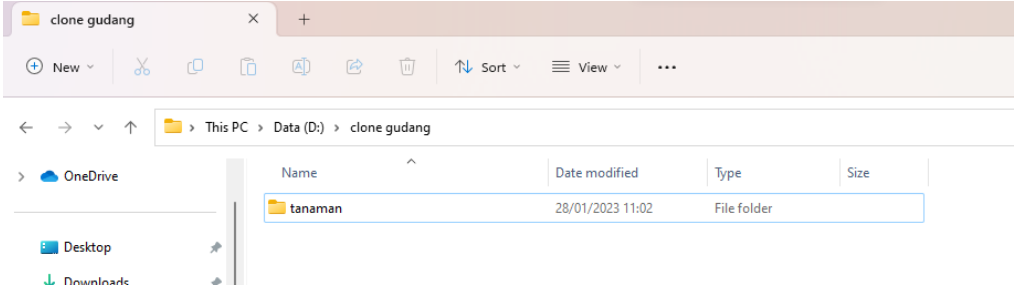
Lalu ketikkan

“git clone https://github.com/MitsaFata/tanaman.git”

Lalu klik enter dan proses clone akan dimulai.

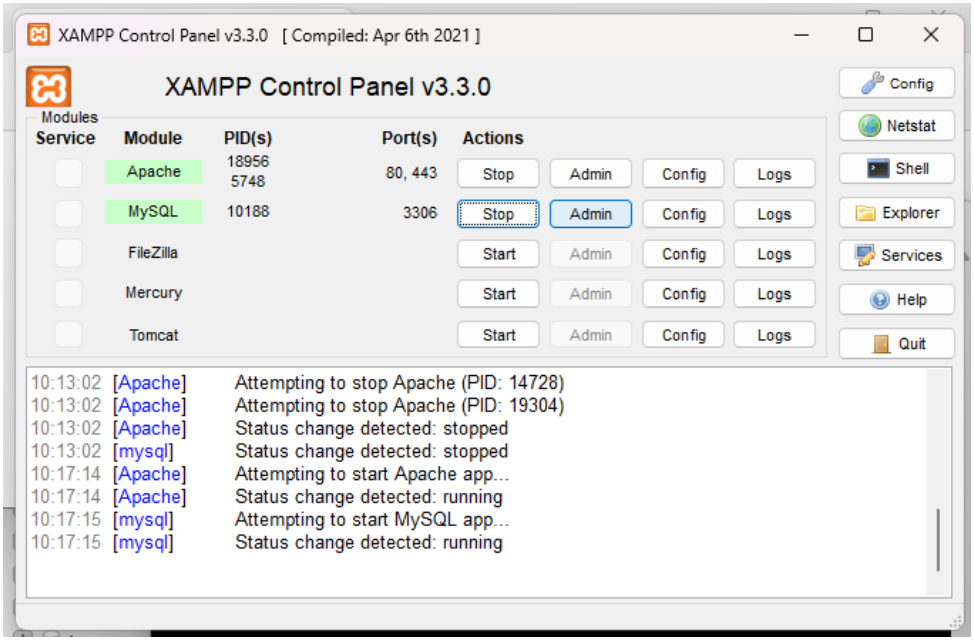


Pada folder yang sudah kita siapkan akan muncul sebuah folder baru yang merupakan repository yang telah kita clone.

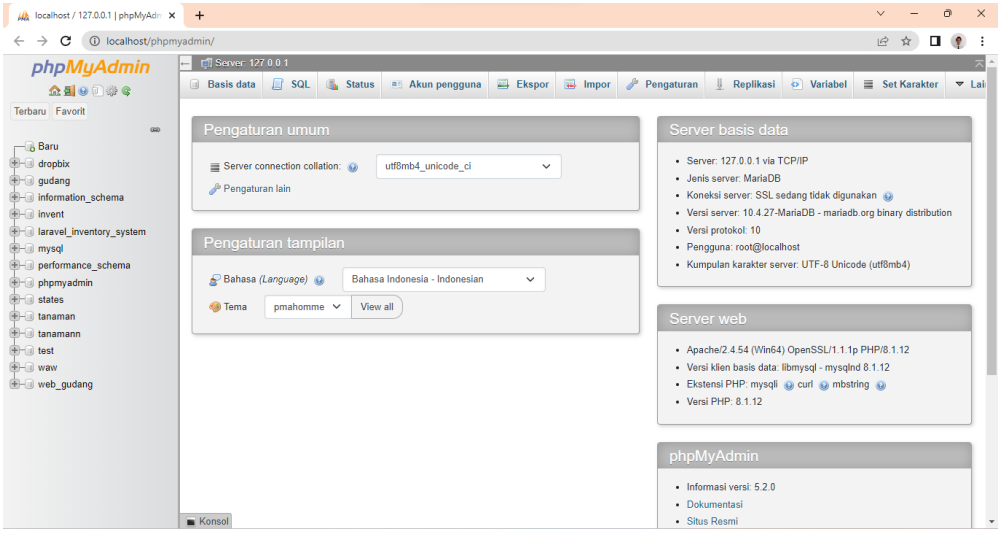


3. Step ketiga adalah membuat database dengan XAMPP.

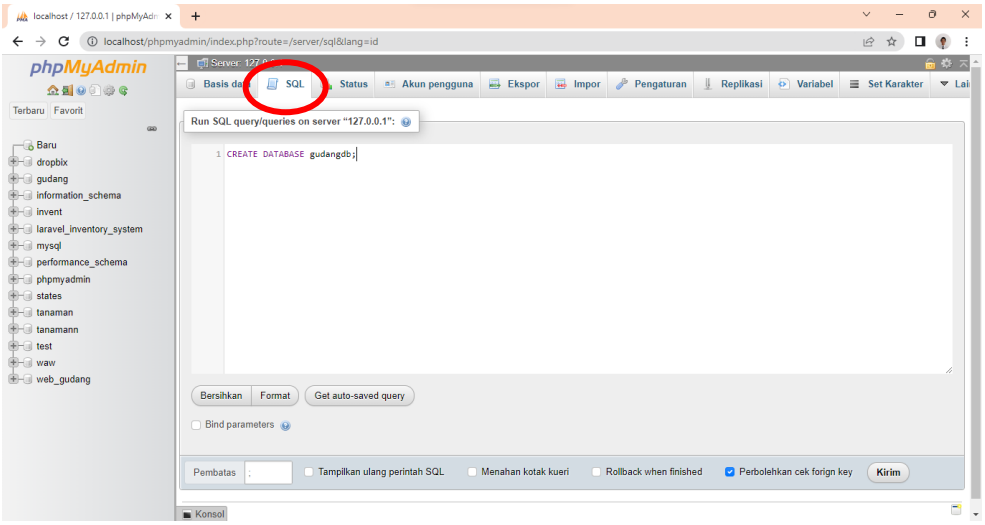
Pertama jalankanlah program xampp, lalu klik start pada module apache dan MySQL, lalu klik admin pada modul MySQL.



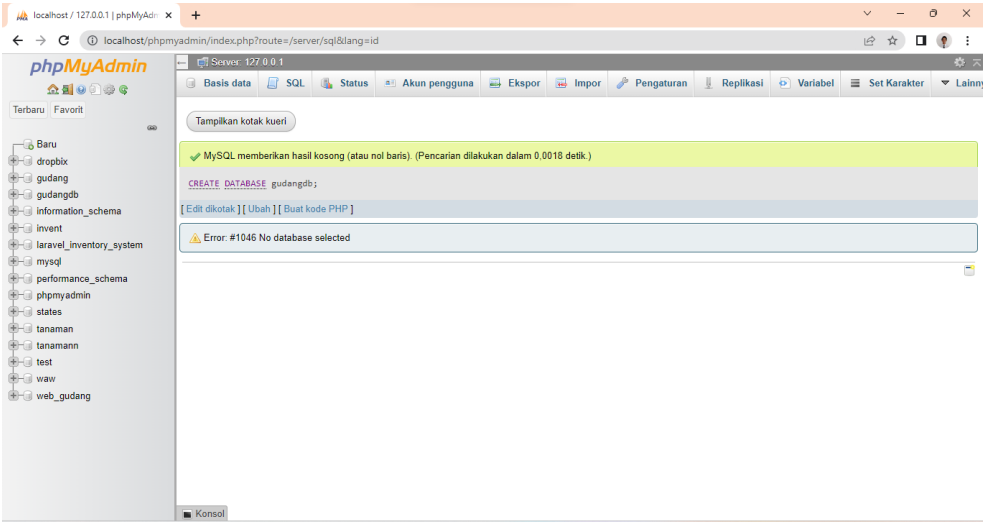
Kemudian kita akan diarahkan ke localhost/phpmyadmin/ pada browser default kita.



Klik pada menu SQL, dan masukkan query “CREATE DATABASE namadatabase;”. Sesuaikan nama databasenya dengan nama yang anda inginkan. Lalu klik Kirim.

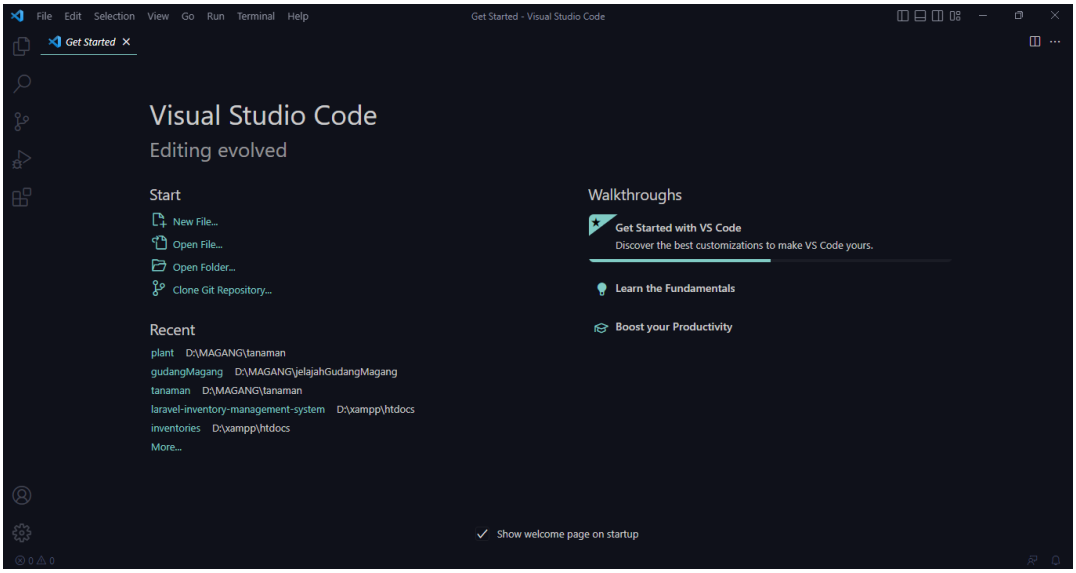


Lalu database baru akan muncul pada tab bagian kiri.

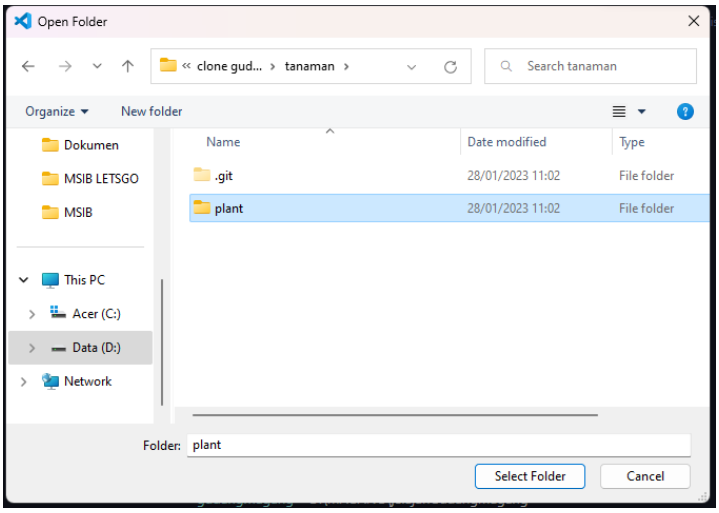


4. Selanjutnya, kita bisa membuka folder proyek Laravel yang sudah kita clone dengan Visual Studio Code.

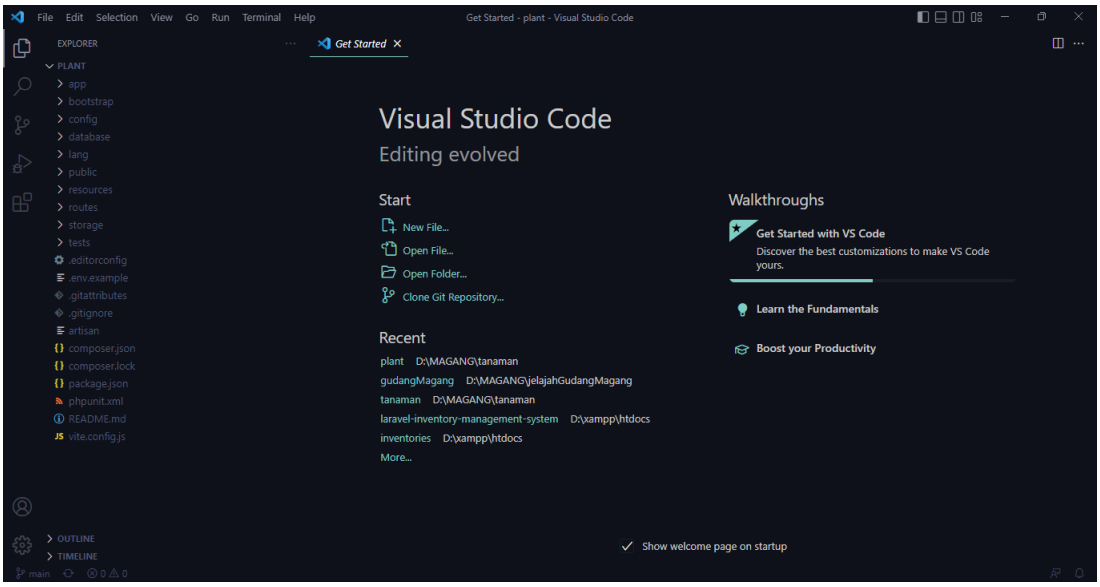
   Setelah tampil halaman get started, pilih opsi open folder.



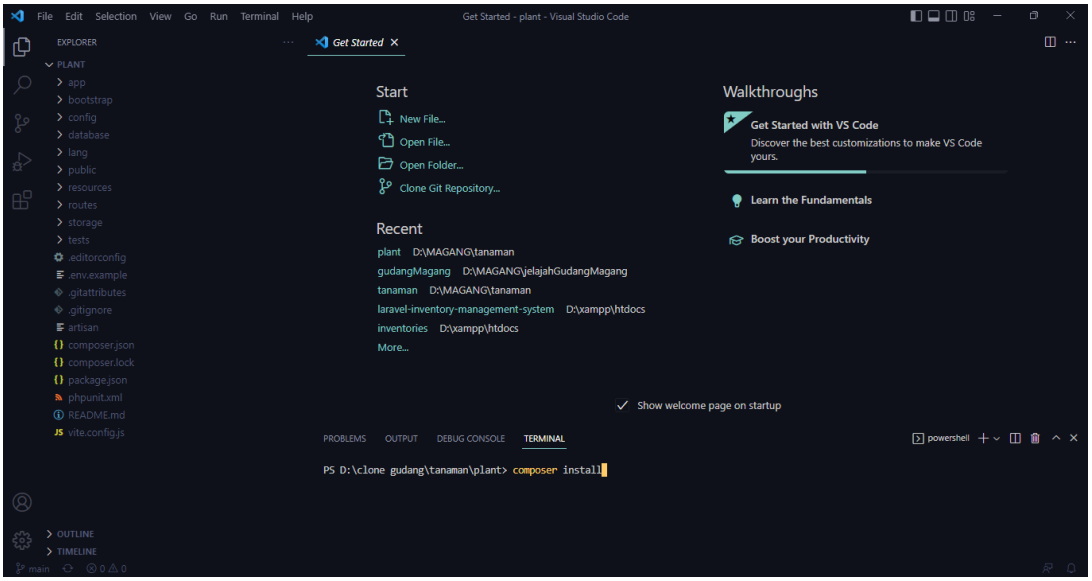
Bukalah folder “plant” dalam folder “tanaman”



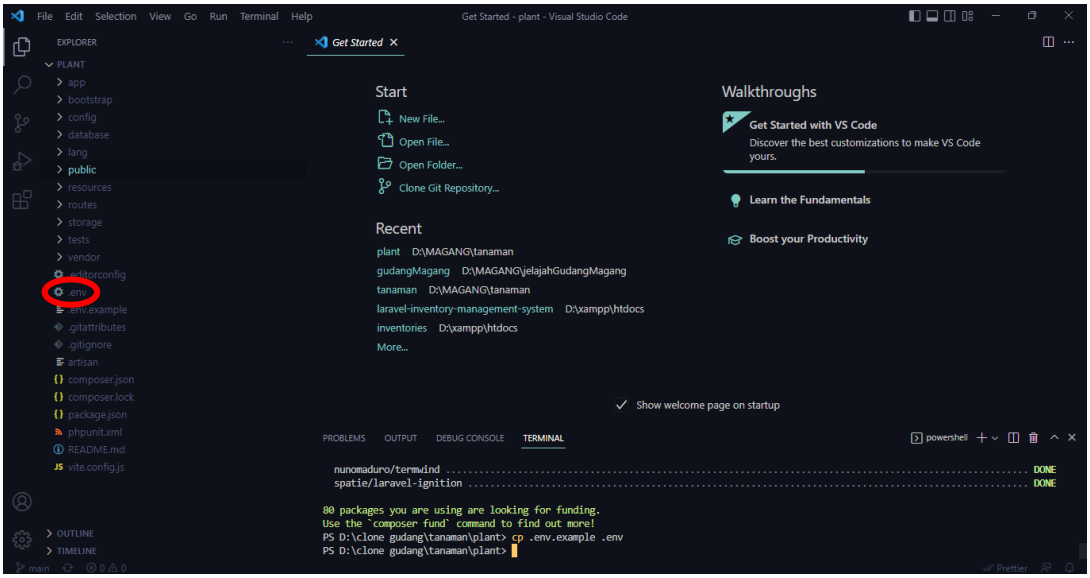
Lalu proyek Laravelnya akan terbuka.



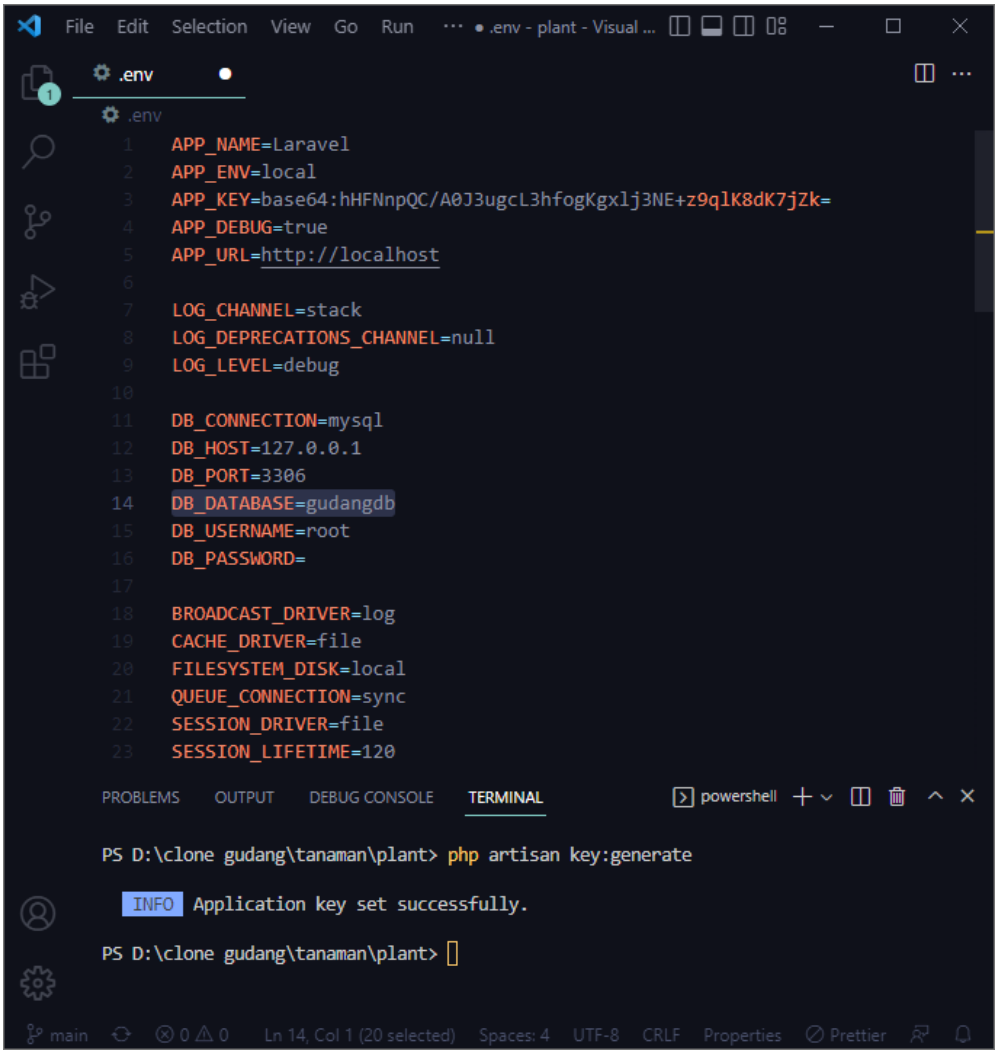
Pada bagian atas klik opsi Terminal > new terminal. Lalu terminal akan terbuka di bagian bawah layar. Disana ketikkan “composer install”. Tekan enter dan tunggu hingga prosesnya selesai.



Setelah prosesnya selesai, ketikkan “cp .env.example .env” dan file env akan muncul.



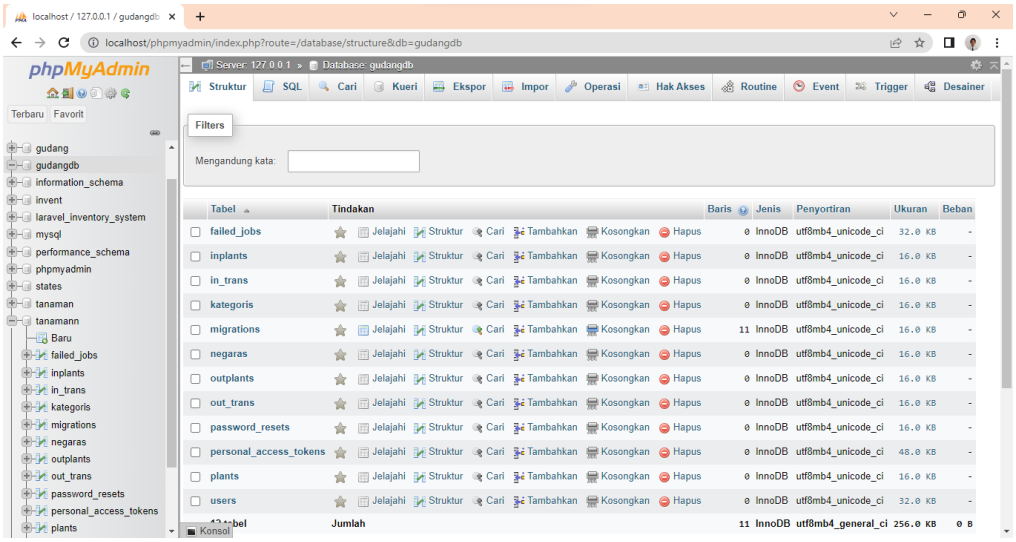
Lalu ketikkan “php artisan key:generate” untuk membuat APP_KEY pada file .env. lalu bukalah file .env dan carilah baris dengan DB_DATABASE.



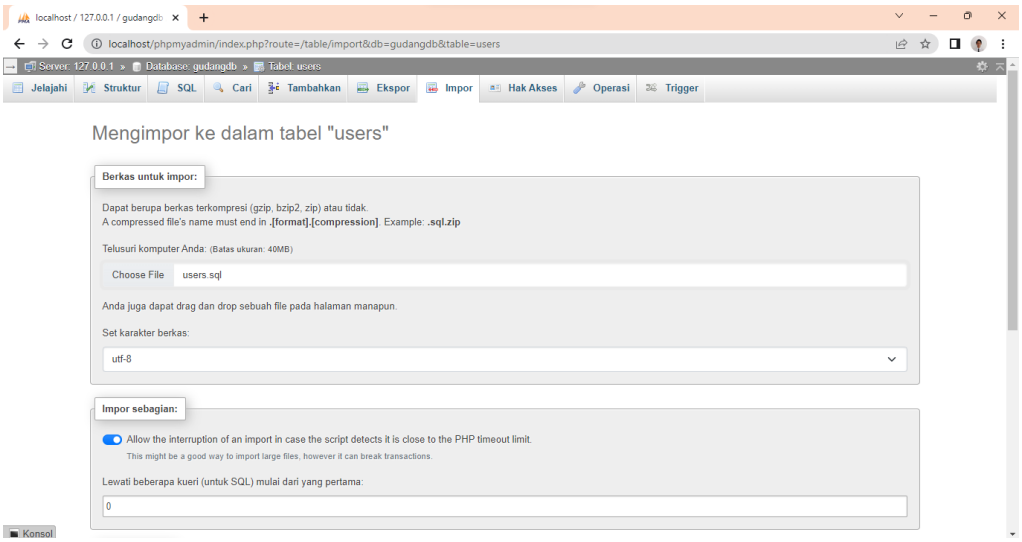
Lalu rubahlah DB_DATABASE dengan nama database yang telah anda buat pada phpMyAdmin, lalu tekan ctrl + s untuk menyimpan perubahan.

Kembali ke Terminal, ketikkan “php artisan migrate” untuk mememigrasi tabel - tabel ke database. Tekan enter dan tunggu hingga selesai.

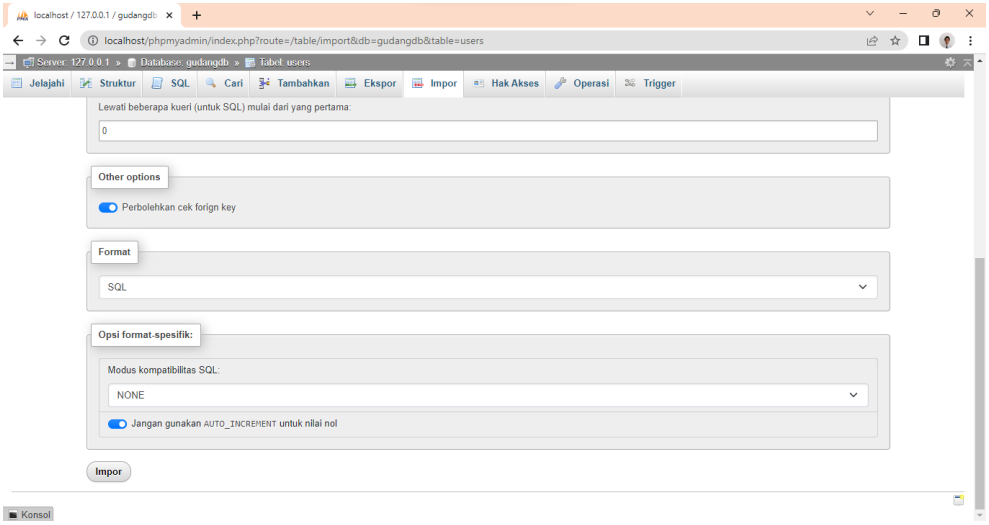
Bisa dilihat pada phpMyAdmin, database yang telah kita buat sudah terisi dengan tabel – tabel.



Kemudian, import data akun superuser dengan file sql yang ada di (isnert folder)

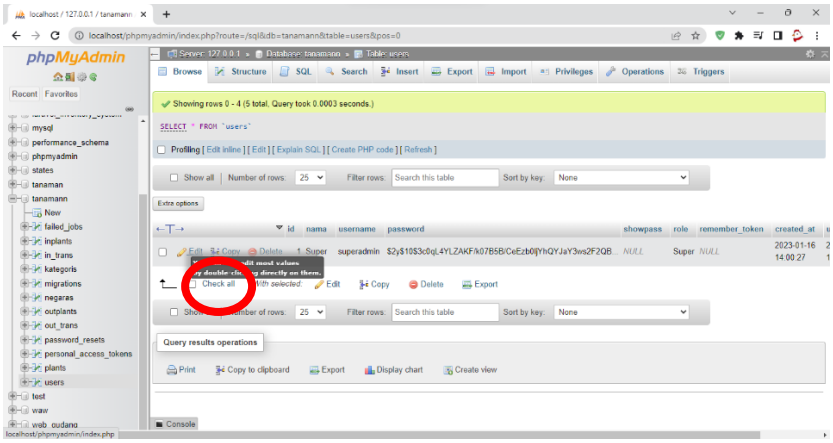


Lalu scroll ke bawah dan klik kirim.

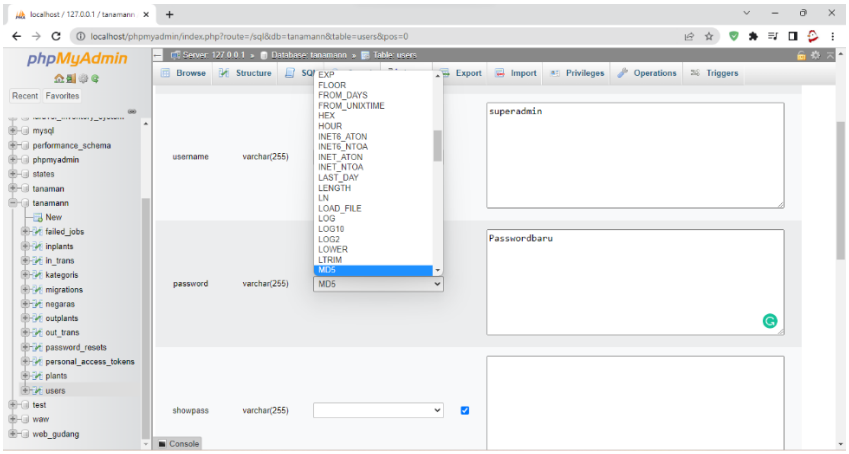


Jika ingin merubah password maka lakukan sebagai berikut :

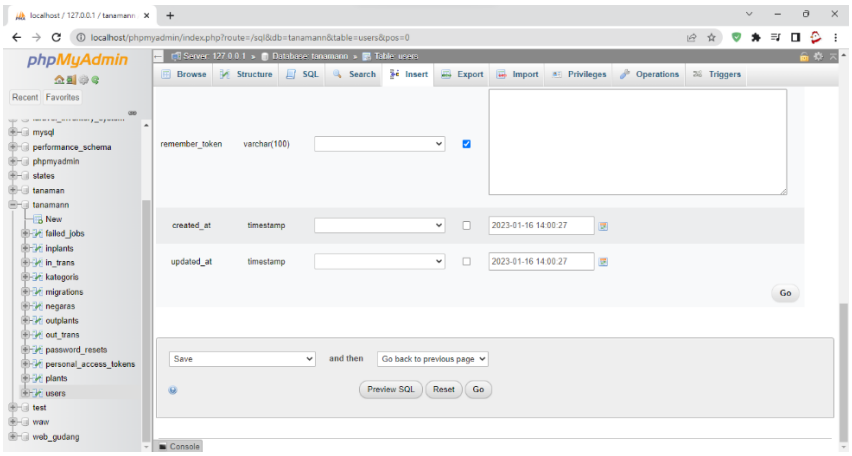
Klik tombol edit pada baris yang ingin diganti passwordnya.



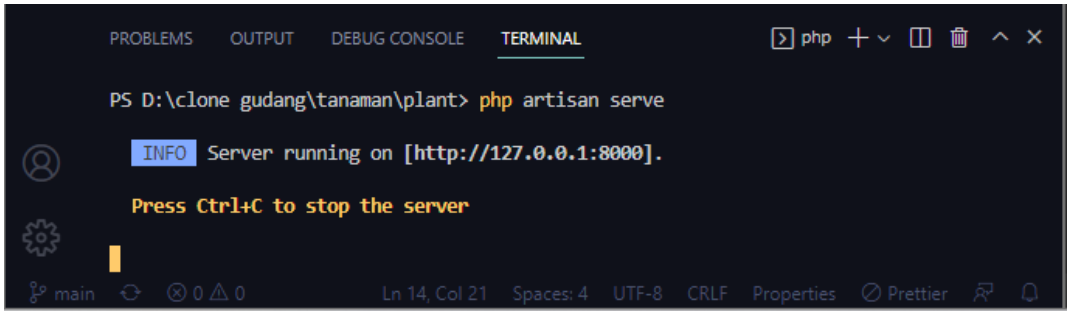
Lalu arahkan ke bagian password dan ganti dengan password baru yang anda inginkan. Rubahlah juga function menjadi MD5.



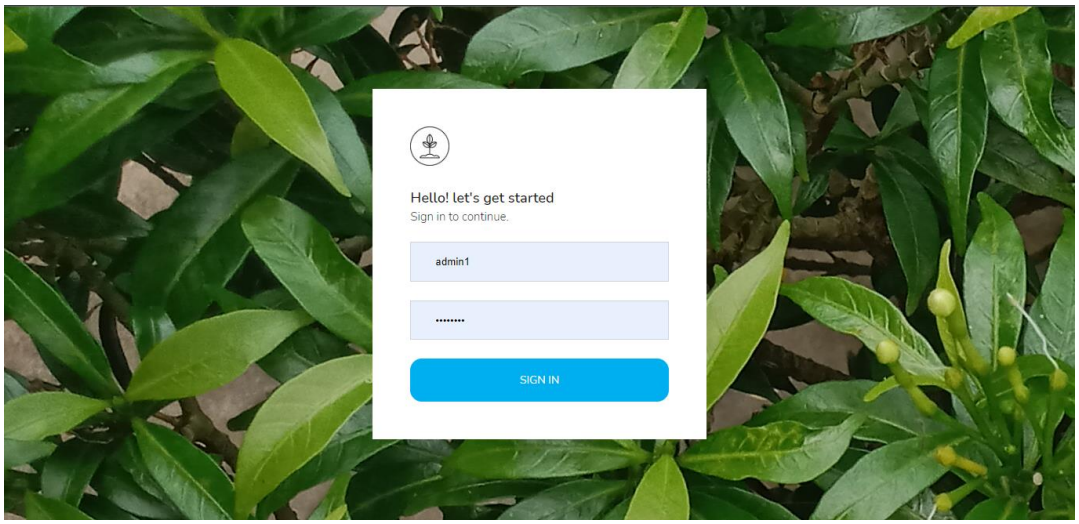
Scroll kebawah dan klik GO.



Selanjutnya, Kembali ke terminal dalam visual studio code, dan ketikkan “php artisan serve” lalu klik enter



Tekan ctrl sambil mengeklik alamat localhost, dan anda akan diarahkan ke website Plant.



Menambah Gudang

Buka App/Http/Controllers/InOut/InController.php

```php
//View In
public function in()
{
    $auto = DB::table('inplants')->select(DB::raw('MAX(RIGHT(id_in,5)) as auto'));
    $kd = "";
    $user = Auth::user()->role;
    $super = Auth::user()->role == "Super";

    if ($user == $super) {
        $ad = 0;
    } else if ($user == "Gudang 1") {
        $ad = 1;
    } else if ($user == "Gudang 2") {
        $ad = 2;
    } else if ($user == "Gudang 3") {
        $ad = 3;
    } else if ($user == "Gudang 4") {
        $ad = 4;
    } else if ($user == "Gudang 5") {
        $ad = 5;
    }
```

Tambahkan Misal Gudang 6

```
} else if ($user == "Gudang 5") {
    $ad = 5;
} else if ($user == "Gudang 6") {
    $ad = 6;
}
```

Setelah itu ganti buka OutController.php

```
//View Out
public function out()
{
    $auto = DB::table('outplants')->select(DB::raw('MAX(RIGHT(id_out,5)) as auto'));
    $kd = "";
    $user = Auth::user()->role;
    $super = Auth::user()->role == "Super";

    if ($user == $super) {
        $ad = 0;
    } else if ($user == "Gudang 1") {
        $ad = 1;
    } else if ($user == "Gudang 2") {
        $ad = 2;
    } else if ($user == "Gudang 3") {
        $ad = 3;
    } else if ($user == "Gudang 4") {
        $ad = 4;
    } else if ($user == "Gudang 5") {
        $ad = 5;
    }
```

Tambahkan Misal Gudang 6

```
} else if ($user == "Gudang 5") {
    $ad = 5;
} else if ($user == "Gudang 6") {
    $ad = 6;
}
```

Pindah ke folder Plant/PlantController.php

```
//View Gudang Admin
public function index()
{
    $user = Auth::user()->role;
    $super = Auth::user()->role == "Super";

    $kategori = DB::table('kategoris')->get();

    $auto = DB::table('plants')->select(DB::raw('MAX(RIGHT(kode_tanaman,5)) as auto'));
    $kd = "";

    if ($user == $super) {
        $ad = "S-0";
    } else if ($user == "Gudang 1") {
        $ad = "A-1";
    } else if ($user == "Gudang 2") {
        $ad = "B-2";
    } else if ($user == "Gudang 3") {
        $ad = "C-3";
    } else if ($user == "Gudang 4") {
        $ad = "D-4";
    } else if ($user == "Gudang 5") {
        $ad = "E-5";
    }
```

\

Tambah gudang misal Gudang 6

```
} else if ($user == "Gudang 5") {
    $ad = "E-5";
} else if ($user == "Gudang 6") {
    $ad = "F-6";
}
```

Pindah ke folder resource/views/User/daftar.blade.php. Cari line seperti berikut dan tambahkan misal gudang 6

```
</div>
<div class="form-group">
    <select class="form-control" id="role" name="role">
        <option value="Gudang 1">Gudang 1</option>
        <option value="Gudang 2">Gudang 2</option>
        <option value="Gudang 3">Gudang 3</option>
        <option value="Gudang 4">Gudang 4</option>
        <option value="Gudang 5">Gudang 5</option>
        <option value="Gudang 6">Gudang 6</option>
    </select>
</div>
```

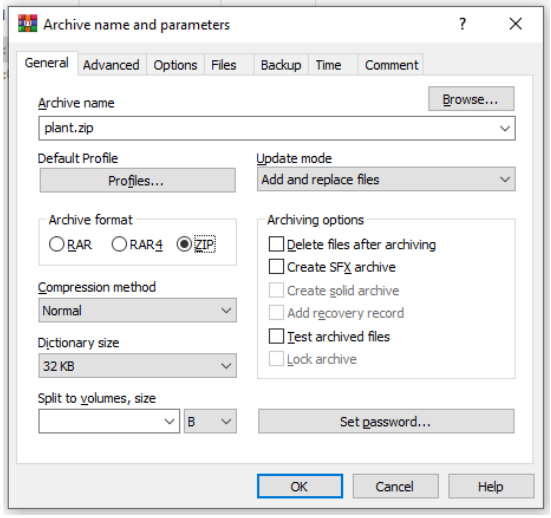
Pada resource/views/index.blade.php. Cari line seperti berikut dan tambahkn missal gudang 6

```
<ul class="nav flex-column sub-menu">
    @if (Auth::user()->role == 'Super')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang</a></li>
    @elseif (Auth::user()->role == 'Gudang 1')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 1</a></li>
    @elseif (Auth::user()->role == 'Gudang 2')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 2</a></li>
    @elseif (Auth::user()->role == 'Gudang 3')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 3</a></li>
    @elseif (Auth::user()->role == 'Gudang 4')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 4</a></li>
    @elseif (Auth::user()->role == 'Gudang 5')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 5</a></li>
    @endif
</ul>
iv>
```

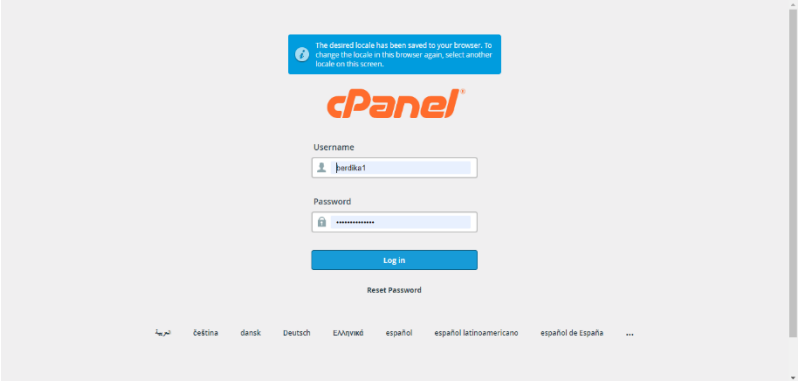
```
    @elseif (Auth::user()->role == 'Gudang 5')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 5</a></li>
    @elseif (Auth::user()->role == 'Gudang 6')
        <li class="nav-item"><a class="nav-link" href="/plant">Gudang 6</a></li>
    @endif
</ul>
```

Hosting

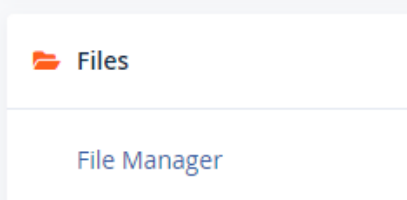
1. Ekstrak File Web ke ZIP



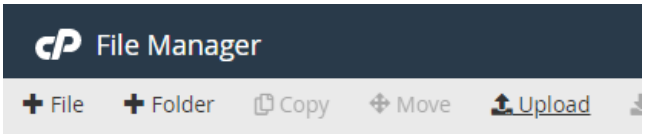
2. Login ke Cpanel



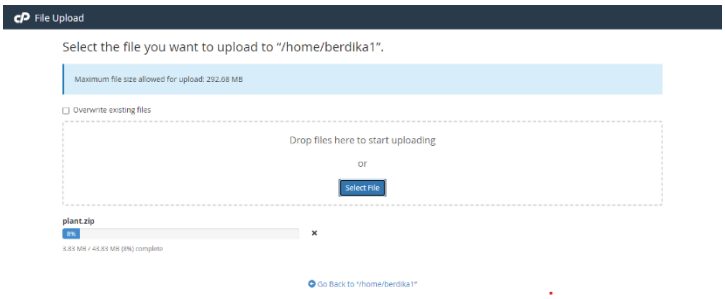
3. Buka file manager



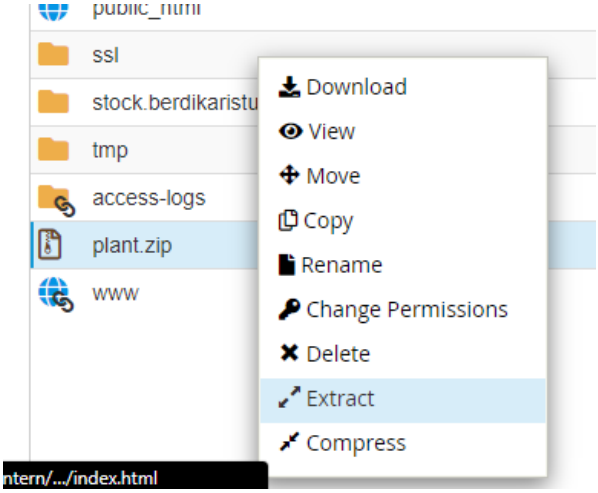
4. Upload file ZIP



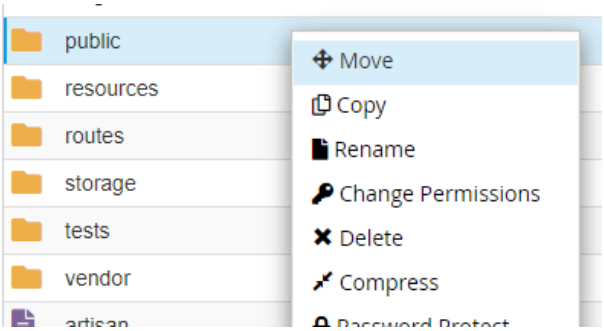
5. Pilih file



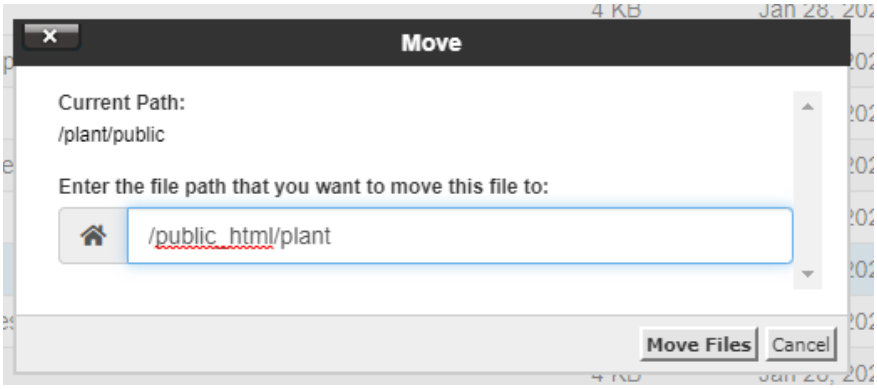
6. Ekstrak file



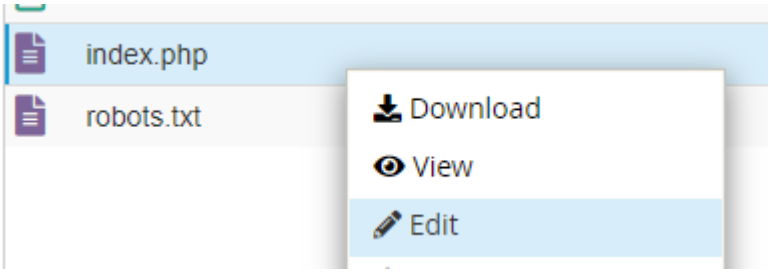
7. Move folder public



8. Ke public_html



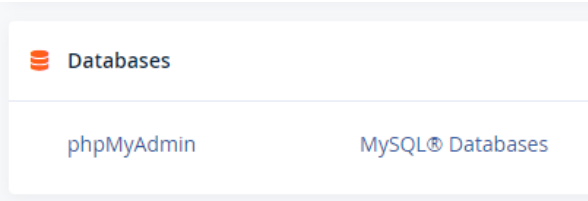
9. Edit file index.php



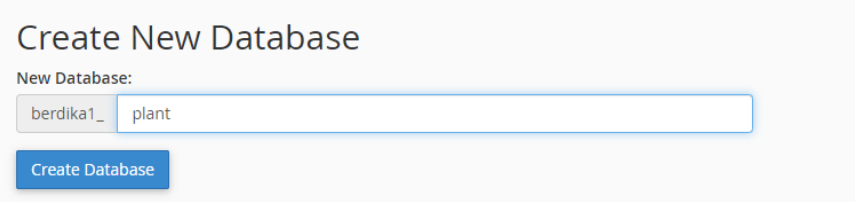
10. Edit agar menuju ke folder file web

```
if (file_exists($maintenance = __DIR__.'/../../plant/storage/framework
    /maintenance.php')) {
    require $maintenance;
}

/*
|--------------------------------------------------------------------------
| Register The Auto Loader
|--------------------------------------------------------------------------
|
| Composer provides a convenient, automatically generated class loader for
| this application. We just need to utilize it! We'll simply require it
| into the script here so we don't need to manually load our classes.
|
*/

require __DIR__.'/../../plant/vendor/autoload.php';

/*
|--------------------------------------------------------------------------
| Run The Application
|--------------------------------------------------------------------------
|
| Once we have the application, we can handle the incoming request using
| the application's HTTP kernel. Then, we will send the response back
| to this client's browser, allowing them to enjoy our application.
|
*/

$app = require_once __DIR__.'/../../plant/bootstrap/app.php';
```

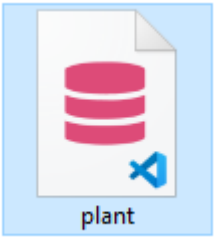
11. Buka database



12. Create new database

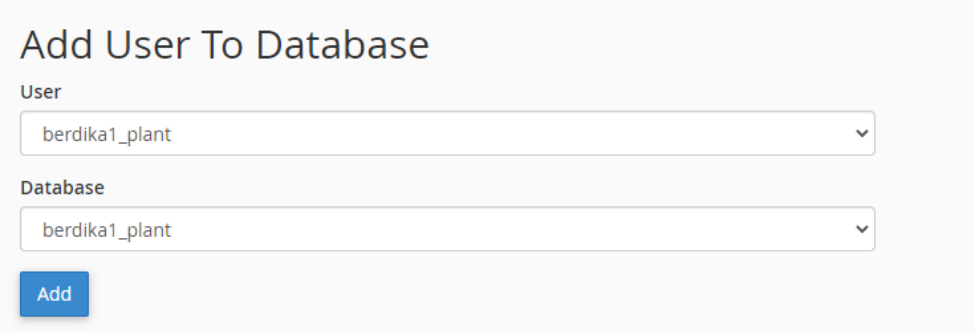


13. Impor sql ke database



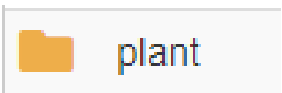
14. Create new user

15. Add user ke database yang sudah dibuat

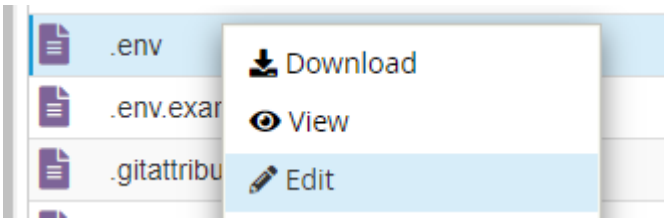


16. Setting privileges user

17. Ke folder web



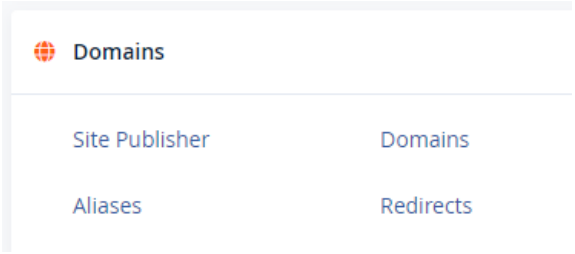
18. Edit env.



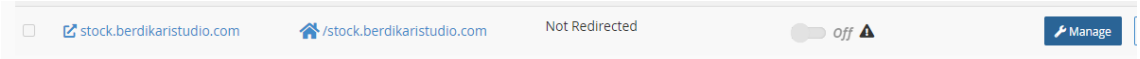
19. Hubungkan ke database dan user

```
10
11  DB_CONNECTION=mysql
12  DB_HOST=127.0.0.1
13  DB_PORT=3306
14  DB_DATABASE=berdika1_plant
15  DB_USERNAME=berdika1_plant
16  DB_PASSWORD=tanduranstok
17
```

20. Buka domains



21. Manage folder domain



22. Arahkan ke public Laravel web