

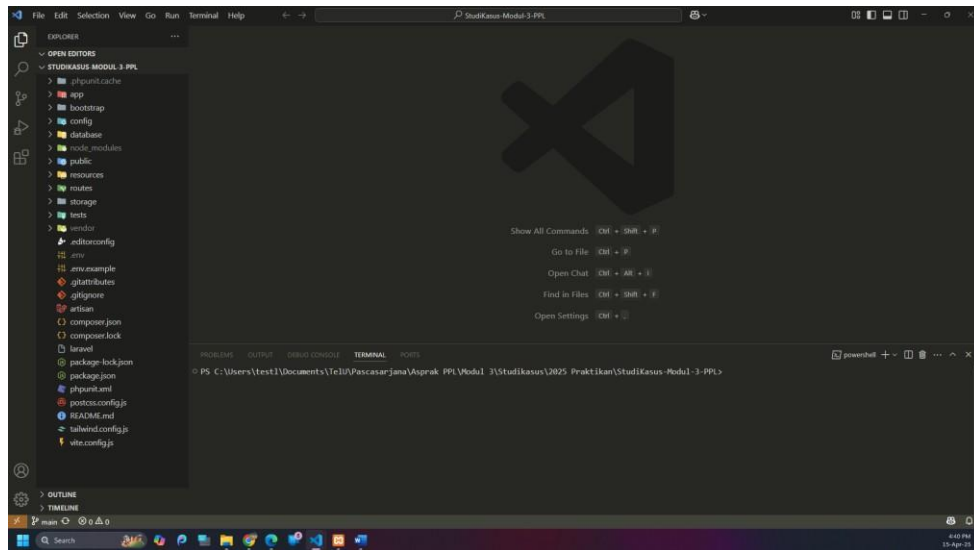
PERATURAN STUDY CASE - INDIVIDU

1. Screenshot **WAJIB** menampilkan **seluruh layar** (termasuk tanggal, dan jam)
2. **Hanya** Screenshot tahap **AKHIR** dari masing masing sub poin (Alphabet).

Contoh:

Clone Repository Modul 3 Automated Testing

- a. Buka link github berikut <https://github.com/alaric2001/StudiKasus-Modul-3-PPL.git>
- b. Lakukan clone repository dan ikuti arahan pada file README.md

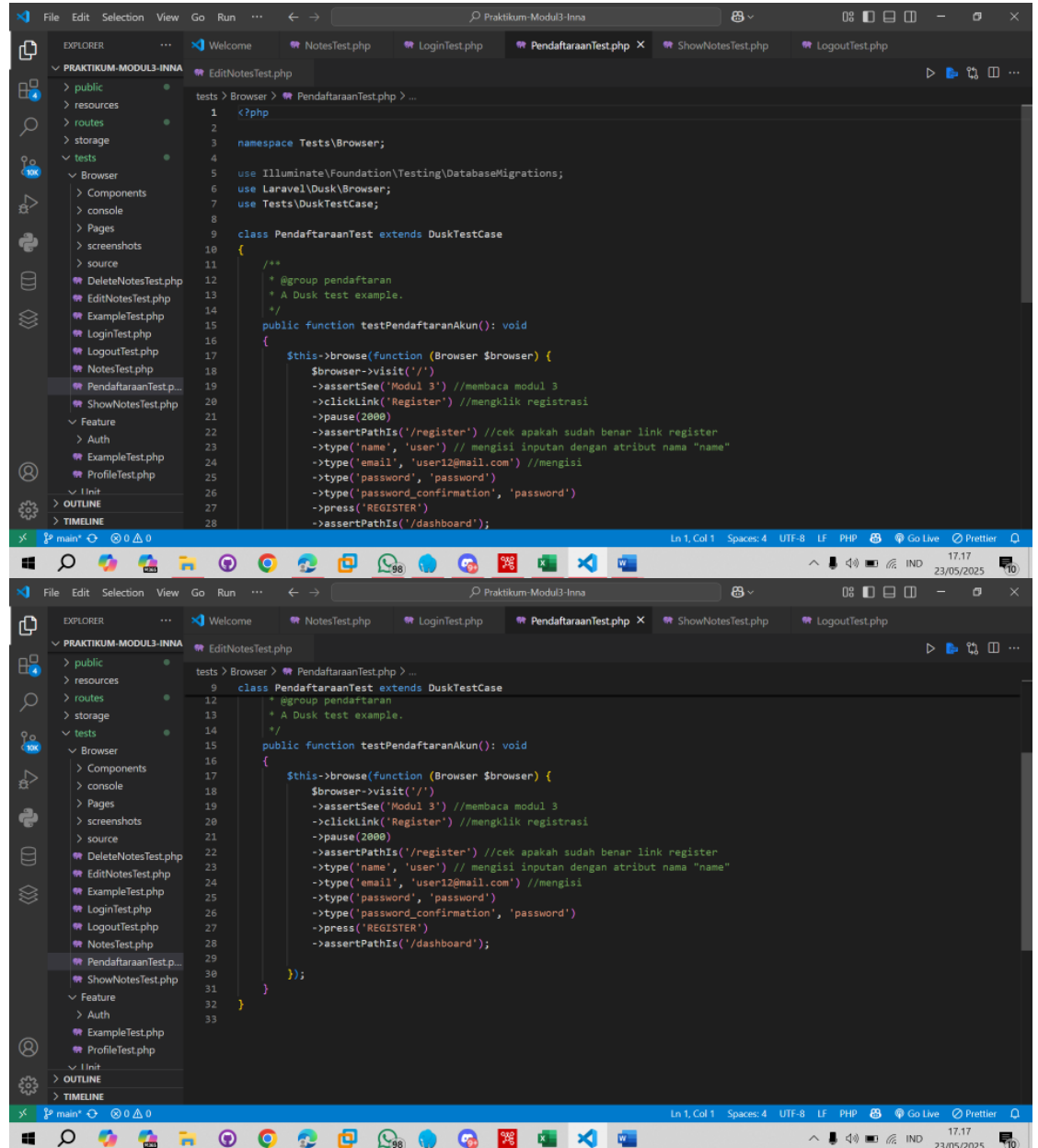


3. Memberikan command atau deskripsi di setiap baris code
Contoh: \$browser->visit('/') //mengunjungi halaman web dengan route "/>
4. Setiap praktikan diwajibkan untuk melakukan membuat repo baru yang berisi proyek Laravel yang telah diclone beserta file-file testingnya.



1. Pengujian Menggunakan Laravel Dusk

- a. Setiap individu membuat testing code untuk fitur **Membuat Akun (Registrasi)**.



The image displays two screenshots of a code editor (VS Code) showing the implementation of a Laravel Dusk test for user registration. The top screenshot shows the initial setup of the test class, and the bottom screenshot shows the completion of the test function.

```
1 <?php
2
3 namespace Tests\Browser;
4
5 use Illuminate\Foundation\Testing\DatabaseMigrations;
6 use Laravel\Dusk\Browser;
7 use Tests\DuskTestCase;
8
9 class PendaftaranTest extends DuskTestCase
10 {
11     /**
12      * @group pendaftaran
13      * A Dusk test example.
14      */
15     public function testPendaftaranAkun(): void
16     {
17         $this->browse(function (Browser $browser) {
18             $browser->visit('/');
19             ->assertSee('Modul 3') //membaca modul 3
20             ->clickLink('Register') //mengklik registrasi
21             ->pause(2000)
22             ->assertPathIs('/register') //cek apakah sudah benar link register
23             ->type('name', 'user') // mengisi inputan dengan atribut nama "name"
24             ->type('email', 'user12@mail.com') //mengisi
25             ->type('password', 'password')
26             ->type('password_confirmation', 'password')
27             ->press('REGISTER')
28             ->assertPathIs('/dashboard');
```

```
9 class PendaftaranTest extends DuskTestCase
10 {
11     /**
12      * @group pendaftaran
13      * A Dusk test example.
14      */
15     public function testPendaftaranAkun(): void
16     {
17         $this->browse(function (Browser $browser) {
18             $browser->visit('/');
19             ->assertSee('Modul 3') //membaca modul 3
20             ->clickLink('Register') //mengklik registrasi
21             ->pause(2000)
22             ->assertPathIs('/register') //cek apakah sudah benar link register
23             ->type('name', 'user') // mengisi inputan dengan atribut nama "name"
24             ->type('email', 'user12@mail.com') //mengisi
25             ->type('password', 'password')
26             ->type('password_confirmation', 'password')
27             ->press('REGISTER')
28             ->assertPathIs('/dashboard');
```

- b. Setiap individu melampirkan hasil dari testing code untuk fitur **Membuat Akun (Registrasi)**.



The screenshot shows the VS Code interface with the file explorer on the left displaying the project structure. The main editor shows the code for `PendaftaranTest.php`, which extends `DuskTestCase` and includes a test method `testPendaftaranAkun()`. The bottom panel shows the output of the test run, indicating that all tests passed successfully.

```

class PendaftaranTest extends DuskTestCase
{
    /**
     * @group pendaftaran
     * A Dusk test example.
     */
    public function testPendaftaranAkun(): void
    {
        $this->browse(function (Browser $browser) {
            // ...
        });
    }
}

```

Tests: 1 failed (3 assertions)
Duration: 15.06s

PS D:\PPL 2025\Praktikum-Modul3-Inna> php artisan dusk --group=pendaftaran
Warning: TTY mode is not supported on Windows platform.

DevTools listening on ws://127.0.0.1:56034/devtools/browser/35c705e4-c375-466e-b134-8668382c8293

PASS Tests\Browser\PendaftaranTest
✓ pendaftaran akun 7.36s

Tests: 1 passed (3 assertions)
Duration: 7.99s

c. Setiap individu **membuat testing code** untuk fitur **Masuk dengan Akun** yang telah dibuat (Login).

The screenshot shows the VS Code interface with the file explorer on the left displaying the project structure. The main editor shows the code for `LoginTest.php`, which extends `DuskTestCase` and includes a test method `testLogin()`. The bottom panel shows the output of the test run, indicating that all tests passed successfully.

```

class LoginTest extends DuskTestCase
{
    /**
     * Tes login.
     * @group Login
     */
    public function testLogin(): void
    {
        $this->browse(function (Browser $browser) {
            $browser->visit('/') // Mengunjungi halaman utama
            ->assertSee('Praktikum PPL') // Memastikan teks "Praktikum PPL" ada
            ->clickLink('Log in') // Mengklik link "Log in"
            ->assertPathIs('/login') // Memastikan berada di halaman login
            ->type('email', 'user12@mail.com') // Mengisi email
            ->type('password', 'password') // Mengisi password
            ->press('LOG IN') // Menekan tombol "LOG IN"
            ->assertPathIs('/dashboard') // Memastikan diarahkan ke halaman dashboard
            ->assertSee('Dashboard'); // Memastikan bahwa teks "Dashboard" muncul
        });
    }
}

```

Ln 1, Col 1 Spaces: 4 UTF-8 LF () PHP Go Live Prettier

d. Setiap individu **melampirkan hasil** dari **testing code** untuk fitur **Masuk dengan Akun** yang telah dibuat (Login).



The screenshot shows a Visual Studio Code editor window with the file explorer on the left displaying a project named 'PRAKTIKUM-MODUL3-INNA'. The main editor area shows the file 'LoginTest.php' with the following code:

```
8 class LoginTest extends DuskTestCase
9 {
10     /**
11      * Test login.
12      * @group Login
13      */
14     public function testLogin(): void
15     {
16         $this->browse(function (Browser $browser) {
17             $browser->visit('/') // Mengunjungi halaman utama
18             ->assertSee('Praktikum PPL') // Memastikan teks "Praktikum PPL" ada
19         });
20     }
21 }
```

The bottom panel shows the terminal output of the test execution:

```
PS D:\PPL 2025\Praktikum-Modul3-Inna> php artisan dusk --group=Login
Warning: TTY mode is not supported on Windows platform.
DevTools listening on ws://127.0.0.1:56147/devtools/browser/d209678c-03a5-4602-8964-148553e15929

PASS Tests\Browser\LoginTest
✓ login

Tests: 1 passed (3 assertions)
Duration: 7.99s
```

e. Setiap individu **membuat testing code** untuk fitur **Membuat Notes**.

The screenshot shows a Visual Studio Code editor window with the file explorer on the left displaying a project named 'STUDIKASUS-MODUL-3-PPL'. The main editor area shows the file 'NotesTest.php' with the following code:

```
1 <?php
2
3 namespace Tests\Browser;
4
5 use Laravel\Dusk\Browser;
6 use Tests\DuskTestCase;
7
8 class NotesTest extends DuskTestCase
9 {
10     /**
11      * Test membuat catatan.
12      * @group Notes
13      */
14     public function testCreateNote(): void
15     {
16         $this->browse(function (Browser $browser) {
17             $browser->visit('/') // Mengunjungi halaman utama website
18             ->clickLink('Log in') // Mengklik link untuk menuju halaman login
19             ->type('email', 'user12@mail.com') // Mengisi input email dengan user terdaftar
20             ->type('password', 'password') // Mengisi input password dengan password user
21             ->press('LOG IN') // Menekan tombol login
22             ->assertPathIs('/dashboard') // Memastikan bahwa user diarahkan ke halaman dashboard
23             ->assertSee('Dashboard') // Memastikan teks "Dashboard" muncul sebagai indikator berhasil login
24
25             ->clickLink('Notes') // Mengklik link "Notes" di dashboard
26             ->assertPathIs('/notes') // Memastikan diarahkan ke halaman /notes
27         });
28     }
29 }
```



The screenshot shows the VS Code editor with the file 'NotesTest.php' open. The code defines a class 'NotesTest' extending 'DuskTestCase' and a method 'testCreateNote()' which performs a series of actions: clicking 'Log in', logging in with 'user12@mail.com' and 'password', clicking 'Notes', clicking 'Create Note', and filling out a form with title 'Catatan Modul 3' and description 'Modul 3 PPL Mengenai Automatic Testing'. The terminal at the bottom shows the command 'php artisan dusk:run-tests' and the output 'PASS Tests\Browser\NotesTest create note'.

- f. Setiap individu **melampirkan hasil dari testing code** untuk fitur **Membuat Notes**.

This screenshot is identical to the one above, showing the 'NotesTest.php' file and the 'testCreateNote()' method. The terminal output at the bottom confirms the successful execution of the test: 'PASS Tests\Browser\NotesTest create note'.

- g. Setiap individu **membuat testing code** untuk fitur **Mengedit Notes**.



The image displays two screenshots of a Visual Studio Code editor window, showing the development of a PHP test using Laravel Dusk. The editor is open to a file named `EditNotesTest.php` within the `tests` directory of a project named `StudiKasus-Modul-3-PPL`.

Top Screenshot: The code shows the initial setup of the `EditNotesTest` class, which extends `DuskTestCase`. It includes namespace declarations and imports for `Illuminate\Foundation\Testing\DatabaseMigrations`, `Laravel\Dusk\Browser`, and `Tests\DuskTestCase`. The `testExample()` method is defined, and the `$this->browse()` function is called with a closure. The closure starts with `$browser->visit('/')` and includes comments for each step.

Bottom Screenshot: The code shows the completed `testExample()` method. It includes the following steps in the closure:

- `$browser->visit('/')`
- `->clickLink('Log in')` // mengeklik login
- `->assertPathIs('/login')` //cek link login
- `->type('email', 'innaasili19@gmail.com')` // mengisi inputan dengan atribut nama email
- `->type('password', 'InnaAsili19')` //mengisi password
- `->press('LOG IN')` // click logging
- `->assertPathIs('/dashboard')` //mengecek apakah sudah masuk dashboard
- `->clickLink('Notes')` //click notes
- `->assertPathIs('/notes')` //cek apakah route notes
- `->clickLink('Edit')` //mengeklik edit
- `->type('title', 'editnote32')` //mengubah title
- `->press('UPDATE')` //click update

- h. Setiap individu melampirkan hasil dari testing code untuk fitur **Mengedit Notes**.



```
File Edit Selection View Go Run ...  
STUDI KASUS-MODUL 3-PPL  
tests > Browser > EditNotesTest.php > PHP Intelephense > EditNotesTest > testExample > Closure  
9 class EditNotesTest extends DuskTestCase  
14  
15 public function testExample(): void  
16 {  
17     $this->browse(function (Browser $browser) {  
18         $browser->visit('/')  
19         ->clickLink('Log in') // mengklik login  
20         ->assertPathIs('/login') //cek link login  
21         ->type('email', 'innaasili19@gmail.com') // mengisi inputan dengan atribut nama email  
22         ->type('password', 'InnaAsili111') //mengisi password  
23         ->press('LOG IN')// click logging  
24         ->assertPathIs('/dashboard') //mengecek apakah sudah masuk dashboard  
25         ->clickLink('Notes')//click notes  
26         ->assertPathIs('/notes') //cek apakah route notes  
27         ->clickLink('Edit') //mengklik edit  
28     }  
29 }  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
DevTools listening on ws://127.0.0.1:50401/devtools/browser/ac9fb87f-d6c3-4215-92e4-0141a5d13bbe  
PASS Tests\Browser\EditNotesTest  
✓ example 6.34s  
Tests: 1 passed (3 assertions)  
Duration: 7.27s  
PS D:\PPL 2025\StudiKasus-Modul-3-PPL>
```

i. Setiap individu membuat testing code untuk fitur **Menampilkan Notes**.

```
File Edit Selection View Go Run ...  
PRAKTIKUM-MODUL3-INNA  
tests > Browser > ShowNotesTest.php > PHP Intelephense > ShowNotesTest  
1 <?php  
2  
3 namespace Tests\Browser;  
4  
5 use Illuminate\Foundation\Testing\DatabaseMigrations;  
6 use Laravel\Dusk\Browser;  
7 use Tests\DuskTestCase;  
8  
9 class ShowNotesTest extends DuskTestCase  
10 {  
11     /**  
12      * A Dusk test example.  
13      * @group displaynote  
14      */  
15     public function testExample(): void  
16     {  
17         $this->browse(function (Browser $browser) {  
18             $browser->visit('/')  
19             ->clickLink('Log in') // mengklik login  
20             ->assertPathIs('/login') //cek link login  
21             ->type('email', 'user12@mail.com') // mengisi inputan dengan atribut nama email  
22             ->type('password', 'password') //mengisi password  
23             ->press('LOG IN')// click logging  
24             ->assertPathIs('/dashboard') //mengecek apakah sudah masuk dashboard  
25             ->clickLink('Notes')//click notes  
26             ->assertPathIs('/notes') //cek apakah route notes  
27             ->clickLink('test') //click note yang tadi sudah dibuat  
28         }  
29     }  
30 }  
Ln 9, Col 20 Spaces: 4 UTF-8 LF PHP Go Live Prettier
```



```
tests > Browser > ShowNotesTest.php > PHP Intelephense > ShowNotesTest
9 class ShowNotesTest extends DuskTestCase
12     * A Dusk test example.
13     * @group displaynote
14     */
15     public function testExample(): void
16     {
17         $this->browse(function (Browser $browser) {
18             $browser->visit('/')
19             ->clickLink('Log in') // mengklik login
20             ->assertPathIs('/login') //cek link login
21             ->type('email', 'user12@mail.com') // mengisi inputan dengan atribut nama email
22             ->type('password', 'password') //mengisi password
23             ->press('LOG IN')// click logging
24             ->assertPathIs('/dashboard') //mengecek apakah sudah masuk dashboard
25             ->clickLink('Notes')//click notes
26             ->assertPathIs('/notes') //cek apakah route notes
27             ->clickLink('test') //click note yang tadi sudah dibuat
28         });
29     }
30 }
31 }
```

- j. Setiap individu melampirkan hasil dari testing code untuk fitur Menampilkan Notes

```
tests > Browser > ShowNotesTest.php > PHP Intelephense > ShowNotesTest > testExample > Closure
9 class ShowNotesTest extends DuskTestCase
15     public function testExample(): void
17     {
18         $this->browse(function (Browser $browser) {
19             ->clickLink('Log in') // mengklik login
20             ->assertPathIs('/login') //cek link login
21             ->type('email', 'user12@mail.com') // mengisi inputan dengan atribut nama email
22             ->type('password', 'password') //mengisi password
23             ->press('LOG IN')// click logging
24             ->assertPathIs('/dashboard') //mengecek apakah sudah masuk dashboard
25             ->clickLink('Notes')//click notes
26             ->assertPathIs('/notes') //cek apakah route notes
27         });
28     }
29 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Tests: 1 failed (3 assertions)
Duration: 7.40s

PS D:\PPL 2025\Praktikum-Modul3-Inna> php artisan dusk --group=displaynote
Warning: TTY mode is not supported on Windows platform.

DevTools listening on ws://127.0.0.1:56289/devtools/browser/e9a73731-6f55-4b0c-8e53-fbdb2864126

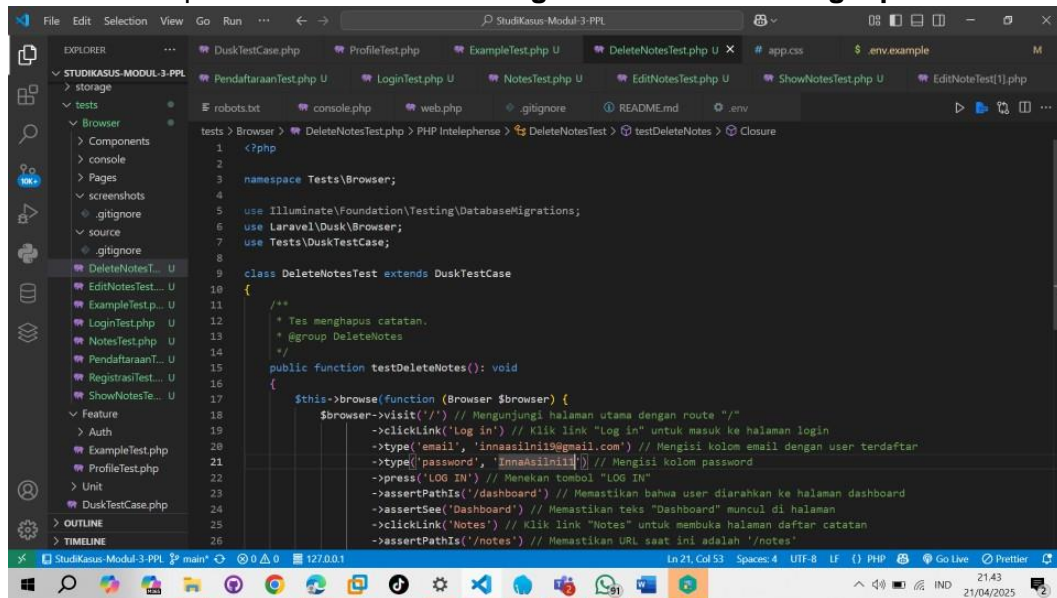
PASS Tests\Browser\ShowNotesTest
✓ example
6.46s

Tests: 1 passed (3 assertions)
Duration: 7.18s

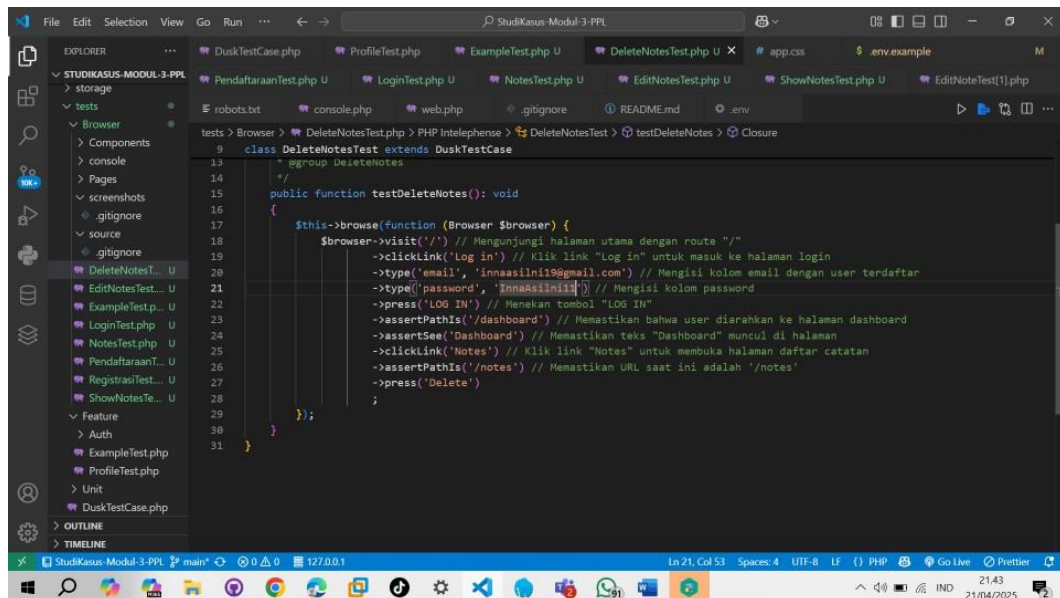
PS D:\PPL 2025\Praktikum-Modul3-Inna>



k. Setiap individu membuat testing code untuk fitur **Menghapus Notes**.

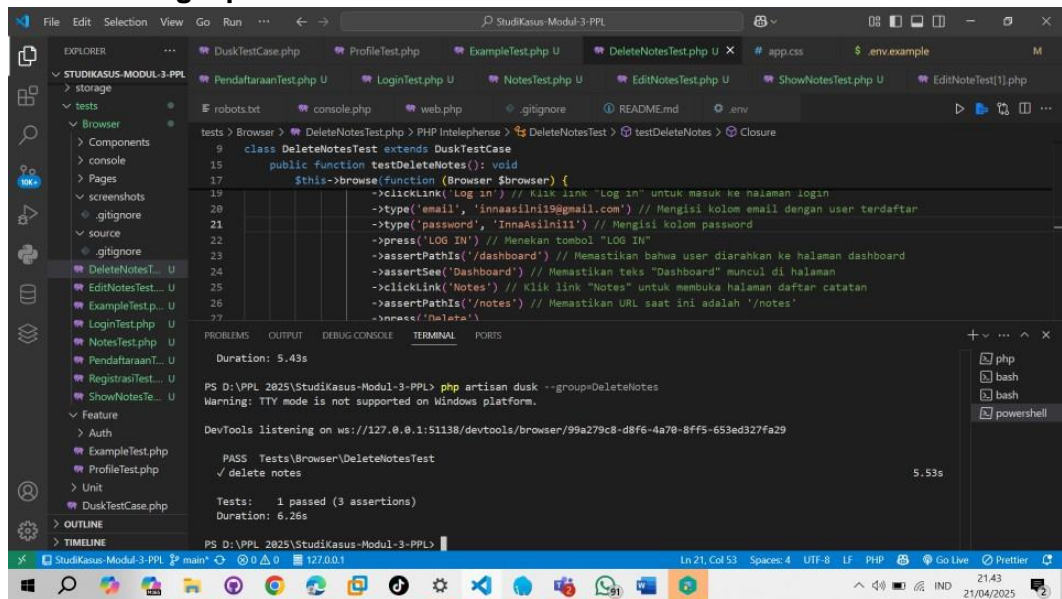


```
1 <?php
2
3 namespace Tests\Browser;
4
5 use Illuminate\Foundation\Testing\DatabaseMigrations;
6 use Laravel\Dusk\Browser;
7 use Tests\Dusk\TestCase;
8
9 class DeleteNotesTest extends Dusk\TestCase
10 {
11     /**
12      * Tes menghapus catatan.
13      * @group DeleteNotes
14      */
15     public function testDeleteNotes(): void
16     {
17         $this->browse(function (Browser $browser) {
18             $browser->visit('/') // Mengunjungi halaman utama dengan route "/"
19             ->clickLink('Log in') // Klik link "Log in" untuk masuk ke halaman login
20             ->type('email', 'innaasili19@gmail.com') // Mengisi kolom email dengan user terdaftar
21             ->type('password', 'InnaAsili111') // Mengisi kolom password
22             ->press('LOG IN') // Menekan tombol "LOG IN"
23             ->assertPathIs('/dashboard') // Memastikan bahwa user diarahkan ke halaman dashboard
24             ->assertSee('Dashboard') // Memastikan teks "Dashboard" muncul di halaman
25             ->clickLink('Notes') // Klik link "Notes" untuk membuka halaman daftar catatan
26             ->assertPathIs('/notes') // Memastikan URL saat ini adalah "/notes"
```



```
1 class DeleteNotesTest extends Dusk\TestCase
2 {
3     /**
4      * @group DeleteNotes
5      */
6     public function testDeleteNotes(): void
7     {
8         $this->browse(function (Browser $browser) {
9             $browser->visit('/') // Mengunjungi halaman utama dengan route "/"
10             ->clickLink('Log in') // Klik link "Log in" untuk masuk ke halaman login
11             ->type('email', 'innaasili19@gmail.com') // Mengisi kolom email dengan user terdaftar
12             ->type('password', 'InnaAsili111') // Mengisi kolom password
13             ->press('LOG IN') // Menekan tombol "LOG IN"
14             ->assertPathIs('/dashboard') // Memastikan bahwa user diarahkan ke halaman dashboard
15             ->assertSee('Dashboard') // Memastikan teks "Dashboard" muncul di halaman
16             ->clickLink('Notes') // Klik link "Notes" untuk membuka halaman daftar catatan
17             ->assertPathIs('/notes') // Memastikan URL saat ini adalah "/notes"
18             ->press('Delete')
19         });
20     }
21 }
```

- l. Setiap individu melampirkan hasil dari testing code untuk fitur Menghapus Notes.



The screenshot shows the VS Code editor with the file `DeleteNotesTest.php` open. The code defines a `DeleteNotesTest` class extending `DuskTestCase` with a `testDeleteNotes` method. The method performs a series of actions: clicking the 'Log in' link, typing the email 'innaaslini19@gmail.com' and password 'InnaAslini11', pressing the 'LOG IN' button, asserting the path is '/dashboard', clicking the 'Notes' link, and asserting the path is '/notes'. The terminal output shows the test passing with 3 assertions in 6.26 seconds.

```
class DeleteNotesTest extends DuskTestCase
{
    public function testDeleteNotes(): void
    {
        $this->browse(function (Browser $browser) {
            ->clickLink('Log in') // Klik link "Log in" untuk masuk ke halaman login
            ->type('email', 'innaaslini19@gmail.com') // Mengisi kolom email dengan user terdaftar
            ->type('password', 'InnaAslini11') // Mengisi kolom password
            ->press('LOG IN') // Menekan tombol "LOG IN"
            ->assertPathIs('/dashboard') // Memastikan bahwa user diarahkan ke halaman dashboard
            ->assertSee('Dashboard') // Memastikan teks "Dashboard" muncul di halaman
            ->clickLink('Notes') // Klik link "Notes" untuk membuka halaman daftar catatan
            ->assertPathIs('/notes') // Memastikan URL saat ini adalah "/notes"
        });
    }
}
```

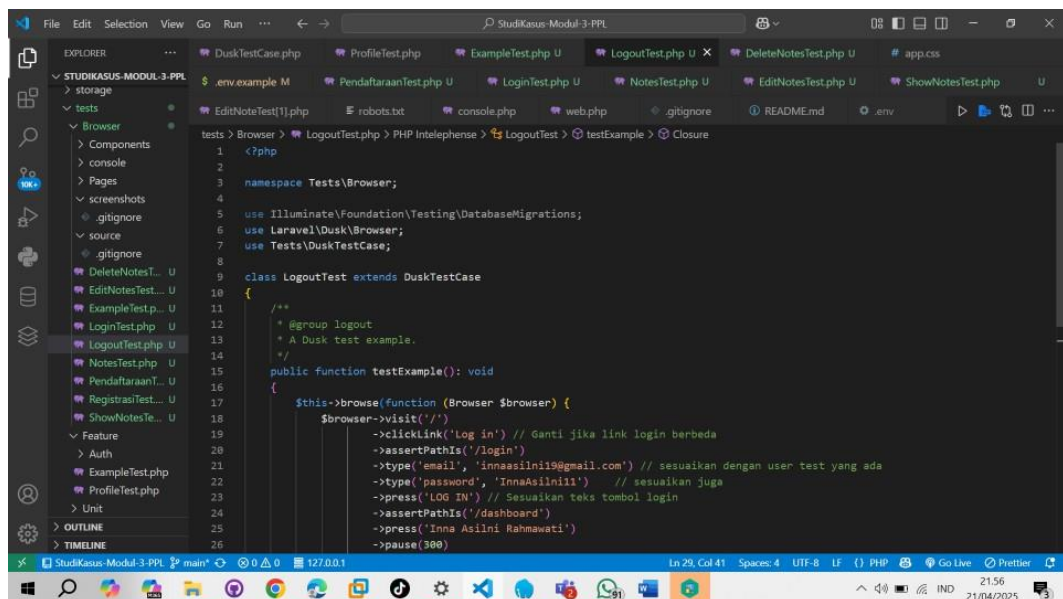
```
PS D:\PPL 2025\StudiKasus-Modul-3-PPL> php artisan dusk --group=DeleteNotes
Warning: TTY mode is not supported on Windows platform.

DevTools listening on ws://127.0.0.1:51138/devtools/browser/99a279c8-d8f6-4a70-8ff5-653ed327fa29

PASS Tests\Browser\DeleteNotesTest
✓ delete notes

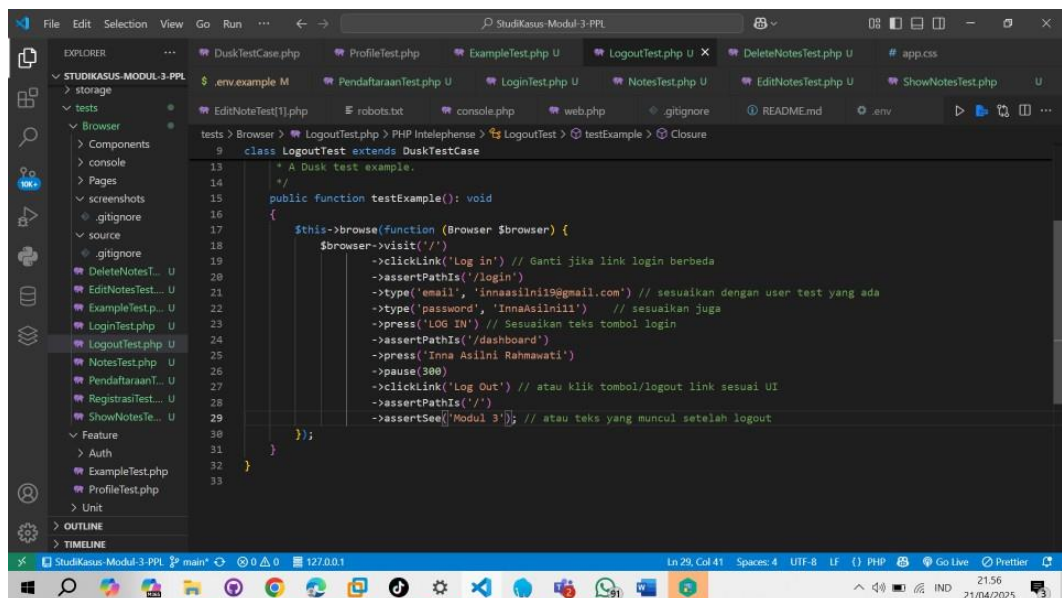
Tests: 1 passed (3 assertions)
Duration: 6.26s
```

- m. Setiap individu membuat testing code untuk fitur Melakukan Logout.

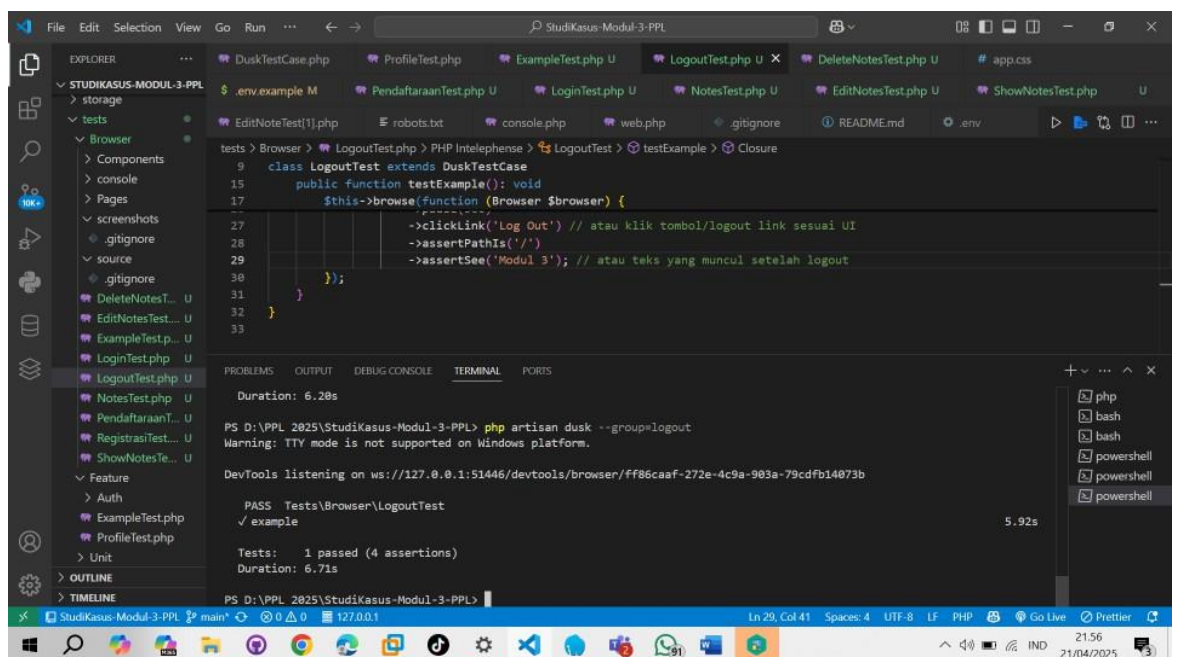


The screenshot shows the VS Code editor with the file `LogoutTest.php` open. The code defines a `LogoutTest` class extending `DuskTestCase` with a `testExample` method. The method performs a series of actions: clicking the 'Log in' link, asserting the path is '/login', typing the email 'innaaslini19@gmail.com' and password 'InnaAslini11', pressing the 'LOG IN' button, asserting the path is '/dashboard', and pressing the 'Inna Aslini Rahmawati' button. The terminal output shows the test passing with 3 assertions in 3.00 seconds.

```
class LogoutTest extends DuskTestCase
{
    /**
     * @group logout
     * A Dusk test example.
     */
    public function testExample(): void
    {
        $this->browse(function (Browser $browser) {
            $browser->visit('/')
            ->clickLink('Log in') // Ganti jika link login berbeda
            ->assertPathIs('/login')
            ->type('email', 'innaaslini19@gmail.com') // sesuaikan dengan user test yang ada
            ->type('password', 'InnaAslini11') // sesuaikan juga
            ->press('LOG IN') // Sesuaikan teks tombol login
            ->assertPathIs('/dashboard')
            ->press('Inna Aslini Rahmawati')
            ->pause(300)
        });
    }
}
```



- n. Setiap individu **melampirkan** hasil dari **testing code** untuk fitur **Melakukan Logout**.



2. Link repository github

<https://github.com/InnaAsilniRahmawati/Praktikum-Modul3-Inna.git>

