

Web Technology



Variable management exercises through branching and arithmetic

Supporting Lecturer:

Ketut Agus Seputra S. ST. M. T

Arranged by:

Tegar Samjoe Indra Laka

2115101032

COMPUTER SCIENCE

INFORMATICS ENGINEERING

TECHNICAL AND VOCATIONAL FACULTY

GANESHA UNIVERCITY OF EDUCATION

2021/2022

Contents Table

A. Problem.	iii
B. Basic Theory	iii
1. PHP.	iii
2. Validation Request Method.	iii
3. Laravel.	iii
C. Descriptions.	iv
1. Visual Studio Code.	iv
A. Structure code.	iv
B. result.	vii
1. Input.	viii
2. Output.	ix
3. If Error.	ix
C. Conclusions.	xi
D. Source Code.	xii

A. Problem.

Silakan adik-adik ikuti latihan sesuai contoh pada video untuk memahami penggunaan kondisi percabangan pada operasi matematika dan manajemen variabel. GUnakan studi kasus berbeda, yang penting sesuai dengan ketentuan.

Ketentuan:

- Antarmuka form input untuk menampung inputan variabel.
- Controller untuk mengolah inputan
- Antarmuka output pada aplikasi untuk menampilkan hasil perhitungan

Kumpulkan dalam pdf laporan dan hasil praktikum.

B. Basic Theory

1. PHP.

PHP is an open-source server-side scripting language that many devs use for web development. It is also a general-purpose language that you can use to make lots of projects, including Graphical User Interfaces (GUIs).

The abbreviation PHP initially stood for Personal Homepage. But now it is a recursive acronym for Hypertext Preprocessor. (It's recursive in the sense that the first word itself is an abbreviation, so the full meaning doesn't follow the abbreviation.)

2. Validation Request Method.

Validation is the process of checking the incoming data. By default, laravel provides the base controller class that uses the ValidatesRequests trait to validate all the incoming Http requests.

3. Laravel.

Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

C. Descriptions.

In this development of this Fibonacci website, I'm using some tools to help me create and getting ready for it, the tools namely:

1. Visual Studio Code.

Visual studio code is a software to write code from various code language and you compile your code on it. I'm use this software to prepare and wrote my code cause thus software so much helping because the extension.

A. Structure code.

1. Fibonacci Controller.

This controller is processing the value that inputted by user. This controller having 2 function they are Function index and function output. Function index is setting up the view for user interface and the function is to count the Fibonacci series and process the value that inputted by user and will be display the output to view using validation request.

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Http\Requests\outputRowValidationRequest;
6  use Illuminate\Http\Request;
7  use Termwind\Components\Dd;
8
9  class FibonacciController extends Controller
10 {
11     public function index()
12     {
13         return view('admin.fibonacci');
14     }
15     public function output(outputRowValidationRequest $request)
16     {
17         //Fibonaaci From Input
18         $row1 = $request->row1;
19         $row2 = $request->row2;
20
21         $fibonacci = [$row1, $row2];
22         $fibonacci[] = $fibonacci[0] + $fibonacci[1];
23
24         for ($i = 2; $i < 10; $i++) {
25             $fibonacci[] = $fibonacci[$i - 1] + $fibonacci[$i - 2];
26         }
27
28         return redirect()->route('fibonacci')->with('success', 'Berhasil menginputkan data dan hasilnya adalah ' . implode(' ', $fibonacci));
29     }
30 }
```

2. Fibonacci Routes.

Fibonacci routes is a route the get and POST the Fibonacci controller to Fibonacci interface.

```
1  <?php
2
3  use App\Http\Controllers\FibonacciController;
4  use App\Http\Controllers\HomeController;
5  use Illuminate\Support\Facades\Route;
6
7  /*
8  |-----
9  | Web Routes
10 |-----
11 |
12 | Here is where you can register web routes for your application. These
13 | routes are loaded by the RouteServiceProvider within a group which
14 | contains the "web" middleware group. Now create something great!
15 |
16 */
17
18 Route::get('/', function () {
19     return view('welcome');
20 });
21
22 Route::get('/fibonacci', function () {
23     return view('admin/fibonacci');
24 }->middleware(['auth', 'verified']->name('admin/fibonacci'));
25
26 Route::get('/fibonacci',[HomeController::class, 'index']);
27 Route::get('/fibonacci', [FibonacciController::class, 'index']->middleware(['auth', 'verified']->name('fibonacci'));
28 Route::post('/fibonacci', [FibonacciController::class,'output']->name('fibonacci'));
29
30 require __DIR__.'/auth.php';
31
```

3. Output Validation Request.

This outputValidationRequest is for authorize the input that requested by user and knowing data that user input is correct or not, if correct will be displaye by 'Success' if not will be displaye by 'Error'.

```
1  <?php
2
3  namespace App\Http\Requests;
4
5  use Illuminate\Foundation\Http\FormRequest;
6
7  class outputRowValidationRequest extends FormRequest
8  {
9      /**
10       * Determine if the user is authorized to make this request.
11       *
12       * @return bool
13       */
14      public function authorize()
15      {
16          return true;
17      }
18
19      /**
20       * Get the validation rules that apply to the request.
21       *
22       * @return array<string, mixed>
23       */
24      public function rules()
25      {
26          return [
27              'row1' => 'required|numeric',
28              'row2' => 'required|numeric',
29          ];
30      }
31  }
32
```

4. Fibonacci Interface.

Fibonnaci interface is the interface that will be displayed to user. This Interface using PHP, Tailwind, and HTML to make the component of the website. And this interface using template-layout to displaying the other interface for the user.

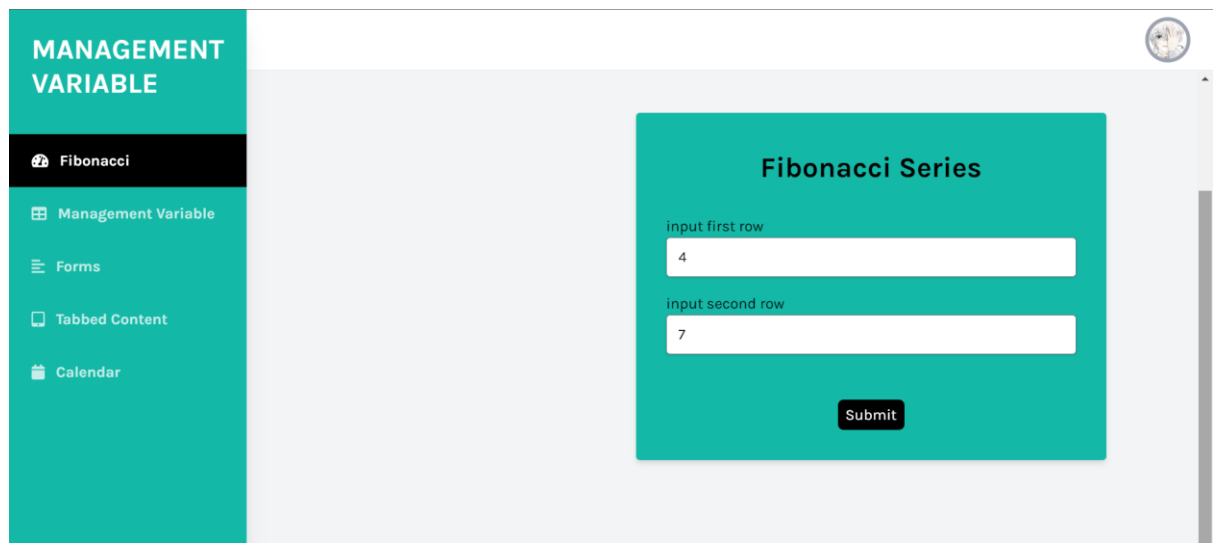
[illegible]

B. result.

This Website using get and POST so the user first time visit the website user will be directed to FibonacciController and the controller will be return to view of the Fibonacci page.

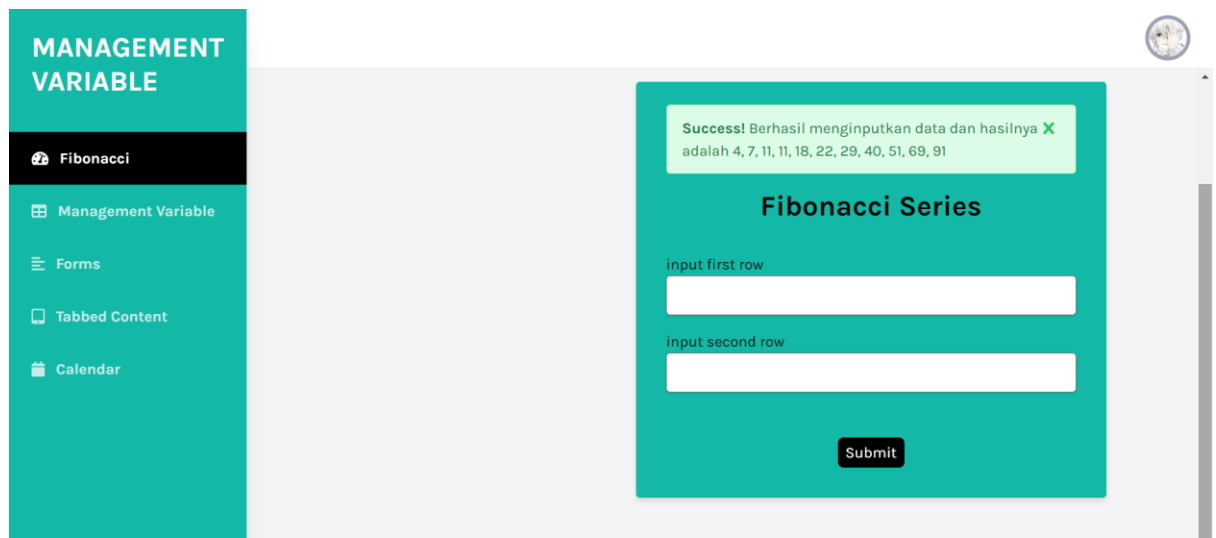
The image is a screenshot of a web application. On the left, there is a teal sidebar with a menu. The menu items are 'Fibonacci' (selected), 'Management Variable', 'Forms', 'Tabbed Content', and 'Calendar'. The top of the sidebar has the text 'MANAGEMENT VARIABLE'. The main content area is light gray. In the center, there is a teal card titled 'Fibonacci Series'. The card contains two input fields: 'input first row' and 'input second row'. Below the input fields is a black button with the text 'Submit'. The top right corner of the page shows a user profile icon.

1. Input.



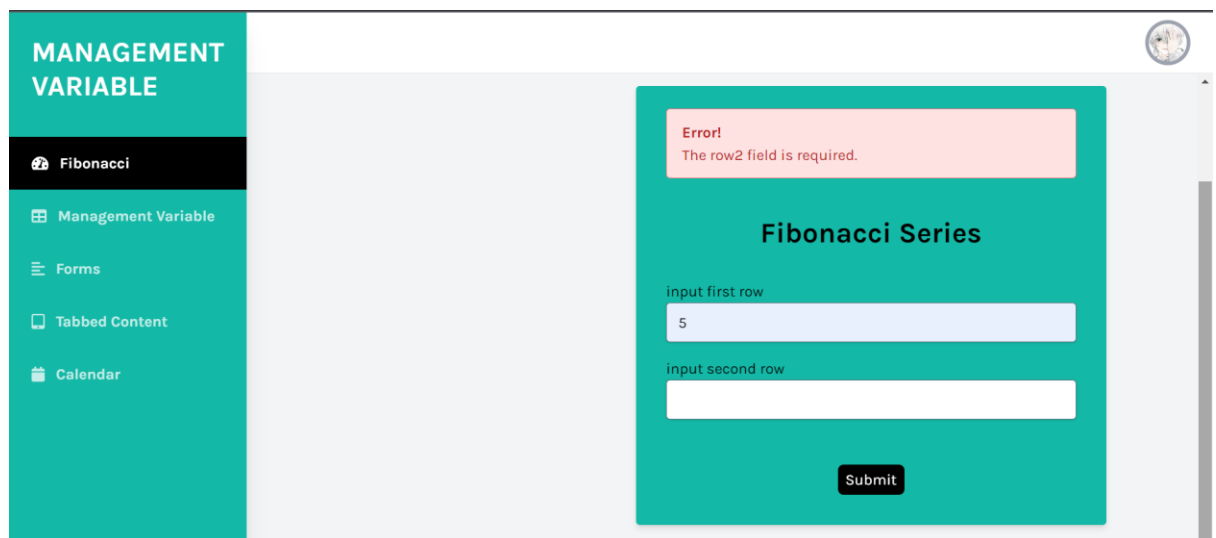
The screenshot displays a web application interface. On the left is a teal sidebar with the text "MANAGEMENT VARIABLE" at the top. Below it, a black bar contains a user icon and the name "Fibonacci". The sidebar lists four menu items: "Management Variable" (with a grid icon), "Forms" (with a list icon), "Tabbed Content" (with a document icon), and "Calendar" (with a calendar icon). The main content area has a light gray background. In the top right corner, there is a circular profile picture of a person. Centered in the main area is a teal card titled "Fibonacci Series". Inside the card, there are two input fields: the first is labeled "input first row" and contains the number "4"; the second is labeled "input second row" and contains the number "7". Below these fields is a black "Submit" button.

2. Output.



The screenshot shows a web application with a teal sidebar on the left and a main content area on the right. The sidebar has a header "MANAGEMENT VARIABLE" and a menu with items: "Fibonacci" (selected), "Management Variable", "Forms", "Tabbed Content", and "Calendar". The main content area has a teal background and contains a "Fibonacci Series" form. At the top of the form is a green success message: "Success! Berhasil menginputkan data dan hasilnya X adalah 4, 7, 11, 11, 18, 22, 29, 40, 51, 69, 91". Below the message are two input fields: "input first row" and "input second row". A black "Submit" button is at the bottom of the form.

3. If Error.



The screenshot shows the same web application as the previous one, but with an error message. The sidebar is identical. The main content area has a teal background and contains the "Fibonacci Series" form. At the top of the form is a red error message: "Error! The row2 field is required." Below the message are two input fields: "input first row" (containing the value "5") and "input second row" (empty). A black "Submit" button is at the bottom of the form.

MANAGEMENT
VARIABLE

Fibonacci

Management Variable

Forms

Tabbed Content

Calendar

Error!
The row1 field is required.

Fibonacci Series

input first row

input second row

7

Submit

MANAGEMENT
VARIABLE

Fibonacci

Management Variable

Forms

Tabbed Content

Calendar

Error!
The row1 field is required.
The row2 field is required.

Fibonacci Series

input first row

input second row

Submit

C. Conclusions.

So in this project of Laravel management variable, we get learning more about PHP and more learning to it's framework Laravel. We learning to use the request method so we can understand the component, controller, route, and interface work. We understand that all of them is complement one and another. The Request is have many method, but for now we will use MVC and Validation method only.

D. Source Code.

Github Link:

- URL Github Profile : <https://github.com/SamjoeTL>
- URL Fibonacci : https://github.com/SamjoeTL/Fibonacci-Series_2115101032