

# Implementasi Filament Form Builder Dalam Sistem Informasi Berbasis Laravel

Sigit Wijanarko

**Abstract**—*Filament Form Builder is a tool that simplifies the creation and management of forms in Laravel-based applications. This journal discusses the installation steps and implementation of Filament Form Builder in a Laravel-based information system. By following the proper installation procedures, developers can enhance efficiency in form creation and management.*

**Intisari**—*Filament Form Builder adalah alat yang mempermudah pembuatan dan pengelolaan formulir dalam aplikasi berbasis Laravel. Jurnal ini membahas langkah-langkah instalasi serta implementasi Filament Form Builder dalam sistem informasi berbasis Laravel. Dengan mengikuti prosedur instalasi yang tepat, pengembang dapat meningkatkan efisiensi dalam pembuatan dan manajemen formulir.*

**Kata Kunci**— *Filament, Form Builder, Laravel.*

## I. PENDAHULUAN

Orang-orang mengisi dan mengirimkan berbagai formulir untuk berbagai keperluan dalam kehidupan sehari-hari mereka. Ketika kita mempertimbangkan antarmuka web, ada beberapa pengontrol standar seperti Kotak teks, Kotak kombo, Tombol radio, dll. Sebagian besar pengontrol ini hanya mengizinkan satu opsi dari grup yang dipilih. Selain itu, pengontrol seperti kotak daftar memiliki opsi pilihan ganda. Beberapa formulir sederhana karena berisi kotak teks dan kontrol yang sangat mudah dipahami. Namun, beberapa kontrol formulir lebih sulit dipahami dan diisi dengan benar oleh pengguna. Dan ketika kita membahas tentang level validasi formulir, level Form, level Bidang adalah level utama dalam memvalidasi formulir [1].

Berkembangnya teknologi informasi saat ini menjadi bagian yang sangat penting diberbagai bidang dalam mendukung proses kegiatan yang dijalani pada bidangnya masing-masing, contohnya pada bidang industri mayoritas telah menggunakan sistem sebagai kinerja atau proses pendukung [2].

User interface merupakan media penjematan antara pengguna dalam berinteraksi dengan sistem sesuai dengan kebutuhan dan keinginan pengguna dengan beberapa pertimbangan yang menjadikan sistem dapat digunakan dengan baik dan tepat [3].

Pengembangan aplikasi web berbasis Laravel, penggunaan formulir merupakan elemen penting untuk mengelola input pengguna. Filament adalah platform administrasi modern yang dirancang untuk mempermudah pengelolaan aplikasi Laravel, termasuk dalam pembuatan dan pengelolaan formulir.

Laravel dapat mempercepat waktu pengembangan dengan mengurangi jumlah kode yang harus ditulis serta

mempermudah integrasi dengan berbagai layanan pihak ketiga. Selain itu, Laravel menyediakan dokumentasi yang komprehensif dan didukung oleh komunitas yang aktif, sehingga membantu dalam pemecahan masalah yang dihadapi pengembang. Dari segi kinerja, aplikasi website yang dibangun dengan laravel menunjukkan responsivitas dan skalabilitas yang baik [10].

Dalam jurnal ini, kami akan membahas langkah-langkah instalasi Filament Form Builder pada proyek Laravel serta perbandingannya dengan pembuatan formulir menggunakan kode native PHP pada proyek Laravel. Perbandingan ini akan membantu untuk lebih memahami keuntungan penggunaan Filament Form Builder dalam hal efisiensi dan kecepatan pengembangan.

## II. KAJIAN LITERATUR

### A. *Form Builder* dalam Pengembangan Web

Pada sebagian besar aplikasi web, atribut kelas entitas secara langsung menentukan konten formulir tampilan yang sesuai. Selain itu, kelas entitas ini sering kali merangkum batasan pada properti terkait. Misalnya, kelas entitas Pengguna mungkin memiliki properti email dengan batasan pada formulir alamat; akibatnya, formulir tampilan untuk membuat/memperbarui pengguna harus menyertakan bidang email dan melakukan validasi pada data yang dikirimkan. Sayangnya, pengembangan formulir tampilan sering kali dilakukan secara manual, proses yang rawan kesalahan dan membosankan. Deteksi kesalahan formulir sangat sulit karena kesalahan hanya muncul saat runtime karena keamanan tipe yang lemah dan mekanisme terbatas untuk verifikasi batasan. Selain itu, modifikasi entitas dapat menyebabkan ketidakkonsistenan dengan formulir yang sesuai [4].

Form builder adalah alat yang dirancang untuk mempermudah pembuatan dan pengelolaan formulir dalam aplikasi web. Formulir adalah salah satu komponen penting dalam aplikasi berbasis web karena digunakan untuk menangkap dan memproses input pengguna [5]. Dalam pengembangannya, form builder seperti Filament menyediakan fitur otomatisasi yang membantu mempercepat proses pembangunan antarmuka pengguna tanpa harus menulis kode HTML dan validasi secara manual.

### B. Efisiensi dalam Pembuatan Formulir

Efisiensi dalam pengembangan perangkat lunak dapat diukur berdasarkan waktu pengembangan, pemeliharaan, dan fleksibilitas sistem [6]. Penggunaan form builder seperti Filament mengurangi waktu yang diperlukan untuk menulis kode secara manual, sehingga meningkatkan produktivitas pengembang. Laravel sebagai framework modern telah mendukung berbagai form builder, termasuk Filament Forms,

*Sistem Informasi, STMIK Antar Bangsa, Kawasan Bisnis CBD Ciledug, Jl. HOS Cokroaminoto No.29-35, RT.001/RW.001, Karang Tengah, Kec. Ciledug, Kota Tangerang, Banten 15157 (telp: 5098-6099; [sgtwijanarko23@gmail.com](mailto:sgtwijanarko23@gmail.com))*

untuk mempermudah pengelolaan data dan integrasi dengan sistem lainnya.

### C. Perbandingan dengan native PHP

Tanpa menggunakan form builder, pengembang harus menulis kode PHP secara manual untuk setiap elemen formulir dan menangani validasi data serta pengelolaan tampilan. Pendekatan manual ini lebih fleksibel tetapi memakan lebih banyak waktu dan rentan terhadap kesalahan dibandingkan pendekatan otomatis dengan form builder [7].

### D. Keamanan dan Validasi Formulir

Dalam proyek pengembangan perangkat lunak, banyak tim memilih untuk menggunakan kerangka kerja untuk membantu dalam proses pengembangan, mengurangi waktu pengiriman aplikasi. Namun, aspek keamanan informasi mungkin tidak diprioritaskan, menghasilkan kerentanan yang dapat dieksploitasi oleh serangan jahat [8].

Validasi input pengguna merupakan aspek penting dalam pengembangan aplikasi web untuk mencegah serangan seperti SQL Injection dan Cross-Site Scripting (XSS). Dengan menerapkan fitur keamanan Laravel secara tepat, aplikasi web dapat dilindungi secara efektif dari berbagai serangan keamanan yang umum [9].

Filament Forms telah terintegrasi dengan mekanisme validasi Laravel yang lebih aman dan lebih mudah digunakan dibandingkan dengan validasi manual menggunakan native PHP.

## III. METODE PENELITIAN

Penelitian ini menggunakan metode eksperimen dengan membandingkan dua teknik pembuatan formulir dalam Laravel, yaitu menggunakan Filament Form Builder dan native PHP. Perbandingan dilakukan berdasarkan kecepatan pengembangan, kemudahan pemeliharaan, serta aspek keamanan dan validasi data.

### A. Analisis dan Perancangan

Pada tahap analisis dan perancangan, formulir input data yang sama didesain menggunakan Filament Form Builder serta native PHP, sehingga memungkinkan perbandingan yang objektif antara kedua metode. Setelah itu, ditentukan beberapa parameter evaluasi yang akan digunakan dalam analisis, seperti jumlah baris kode yang diperlukan, waktu pengembangan yang dibutuhkan, serta efektivitas validasi data. Dengan pendekatan ini, penelitian dapat mengidentifikasi kelebihan dan kekurangan masing-masing metode secara lebih sistematis.

### B. Implementasi

Implementasi merupakan sebuah penempatan ide, konsep, kebijakan, atau inovasi dalam suatu tindakan praktis sehingga memberikan dampak, baik berupa perubahan pengetahuan, keterampilan maupun nilai dan sikap [4].

Pada bagian implementasi ini, pertama-tama dilakukan pembangunan formulir menggunakan Filament Form Builder, yang merupakan alat untuk membangun antarmuka formulir secara efisien. Selanjutnya, formulir dibangun menggunakan native PHP dengan proses validasi manual untuk membandingkan pendekatan yang lebih tradisional. Setelah

kedua formulir selesai dibangun, langkah selanjutnya adalah mengukur waktu yang diperlukan untuk menyelesaikan masing-masing formulir, guna mengevaluasi efisiensi dan kecepatan kedua metode tersebut.

### C. Evaluasi dan Analisis Data

Pada bagian evaluasi dan analisis data, langkah yang dilakukan adalah menghitung dan membandingkan jumlah baris kode yang diperlukan pada kedua metode yang diuji. Selanjutnya, dilakukan uji coba untuk mengevaluasi beberapa aspek penting, seperti keamanan, responsivitas, dan validitas data yang dihasilkan oleh masing-masing metode. Terakhir, analisis mendalam dilakukan untuk menilai tingkat kesalahan atau potensi celah keamanan yang ada dalam kedua metode tersebut, guna memastikan keandalan dan efisiensinya dalam berbagai kondisi.

## IV. HASIL DAN PEMBAHASAN

Dengan mengikuti prosedur instalasi yang benar, Filament Form Builder dapat diterapkan secara optimal dalam sistem informasi berbasis Laravel. Alat ini memberikan kemudahan dalam pembuatan formulir yang dapat disesuaikan sesuai kebutuhan pengguna. Selain itu, integrasi dengan Tailwind CSS memastikan tampilan formulir yang responsif dan modern.

### A. Metode Instalasi dan Pembuatan Formulir

Penelitian ini membandingkan efisiensi pembuatan formulir input data nama dan email menggunakan Filament Form Builder dan native PHP. Formulir yang dibuat memiliki dua field utama: Nama dan Email, serta dilengkapi dengan validasi dasar.

Instalasi instalasi Filament Form Builder, sistem harus memenuhi persyaratan yang dibutuhkan, seperti PHP versi 8.1 ke atas, Laravel 10.0 atau lebih tinggi, dan Livewire 3.0 atau lebih tinggi.

Mulai dengan menambahkan Paket Form Builder: Instal melalui Composer.

```
composer require filament/forms:"^3.2" -W
```

Pada proyek Laravel baru, instalasi Filament bersama dengan form builder.

```
php artisan filament:install --scaffold --forms  
npm install  
npm run dev
```

Lalu, menerapkan kode implementasi formulir menggunakan Filament Form Builder dalam Laravel.

```

use Filament\Forms\Components\TextInput;
use Filament\Forms\Form;
use Filament\Resources\Pages\CreateRecord;

class CreateUser extends CreateRecord
{
    protected static string $resource = UserResource::class;

    protected function form(Form $form): Form
    {
        return $form->schema([
            TextInput::make('name')
                ->required()
                ->maxLength(255),
            TextInput::make('email')
                ->email()
                ->required()
                ->unique('users', 'email'),
        ]);
    }
}

```

Kode Filament, sederhana & terstruktur. Formulir dibuat dengan hanya mendefinisikan field yang dibutuhkan. Validasi Otomatis: Filament langsung menangani validasi seperti required, email format, dan unique constraint. Integrasi dengan Database: Formulir dapat langsung tersimpan ke tabel users tanpa kode tambahan.

Jika formulir dibuat dengan native PHP (tanpa Filament), kodenya menjadi lebih panjang. Karena adanya HTML Formulir (Blade Template Laravel) dan Validasi serta Penyimpanan (Controller Laravel dengan native PHP).

Mulai dengan membuat Blade Template Laravel.

```

<form method="POST" action="{{ route('user.store') }}">
    @csrf
    <label for="name">Nama:</label>
    <input type="text" name="name" required>

    <label for="email">Email:</label>
    <input type="email" name="email" required>

    <button type="submit">Submit</button>
</form>

```

Lalu membuat Controller Laravel dengan native PHP.

```

use Illuminate\Http\Request;
use App\Models\User;

class UserController extends Controller
{
    public function store(Request $request)
    {
        $request->validate([
            'name' => 'required|max:255',
            'email' => 'required|email|unique:users,email',
        ]);

        User::create([
            'name' => $request->input('name'),
            'email' => $request->input('email'),
        ]);

        return redirect()->back()->with('success', 'User berhasil ditambahkan');
    }
}

```

Dengan native PHP, kode lebih panjang & kompleks: Harus membuat form HTML secara manual, menambahkan validasi di controller, serta menangani penyimpanan data. Validasi Harus Ditulis Terpisah: Developer perlu memastikan semua validasi didefinisikan di controller, berpotensi meningkatkan kompleksitas pemeliharaan kode. Kurang Terintegrasi: Tidak memiliki dukungan antarmuka admin yang langsung siap pakai seperti pada Filament.

## B. Perbandingan Efisiensi

Berdasarkan implementasi di atas, Filament Form Builder menawarkan pendekatan yang lebih efisien dibandingkan dengan native PHP dalam pembuatan dan manajemen formulir input data. Filament mengurangi jumlah kode yang harus ditulis, mengotomatiskan validasi, dan lebih mudah dalam pemeliharaan.

Sebaliknya, dengan native PHP, pengembang harus menangani setiap aspek formulir secara manual, mulai dari pembuatan HTML, validasi di controller, hingga penyimpanan data, yang meningkatkan kompleksitas dan potensi kesalahan dalam pengelolaan kode.

Dengan demikian, penggunaan Filament Form Builder direkomendasikan untuk meningkatkan efisiensi dalam pengembangan aplikasi Laravel, terutama dalam proyek yang memerlukan banyak formulir dengan validasi dan penyimpanan data yang kompleks.

Tabel 1. Perbandingan Efisiensi Pembuatan Formulir

Aspek	Filament Form Builder	Native PHP
Kecepatan Pengembangan	Sangat cepat, menggunakan form builder terintegrasi dengan Laravel. Mengurangi waktu untuk menulis kode secara manual.	Lambat, karena memerlukan penulisan kode untuk setiap elemen form secara manual dan validasi.
Kemudahan Penggunaan	Memudahkan dengan antarmuka pengguna yang terstruktur dan dokumentasi yang jelas.	Memerlukan pemahaman mendalam tentang form, validasi, dan pengelolaan data, sering kali dengan sedikit dukungan visual.
Tampilan & Desain	Menggunakan Tailwind CSS secara default untuk desain responsif dan modern.	Memerlukan usaha lebih untuk menerapkan desain responsif, biasanya mengandalkan CSS kustom atau framework lain.
Pemeliharaan	Mudah dikelola dan diperbarui dengan integrasi Filament yang sering mendapat pembaruan.	Memerlukan pembaruan manual dan modifikasi kode untuk memperbaiki bug atau menambah fitur baru.

Keamanan	Filament mengintegrasikan fitur keamanan Laravel dan pengelolaan form yang lebih aman secara otomatis.	Keamanan tergantung pada pengembang untuk mengimplementasikan validasi dan perlindungan yang tepat.
Fleksibilitas	Filament menyediakan berbagai komponen siap pakai, meskipun beberapa kustomisasi lebih terbatas pada paket ini.	Sangat fleksibel, karena pengembang memiliki kendali penuh atas setiap elemen formulir.

## V. KESIMPULAN

Penggunaan Filament Form Builder jauh lebih efisien dibandingkan dengan pembuatan dan manajemen formulir menggunakan native PHP. Filament menyediakan solusi yang lebih cepat, lebih aman, dan lebih mudah untuk mengelola formulir dalam aplikasi Laravel, sementara native PHP memerlukan lebih banyak waktu untuk menulis kode, memelihara form, dan mengelola desain. Dengan demikian, Filament sangat disarankan bagi pengembang yang ingin meningkatkan efisiensi dan kualitas pengembangan aplikasi Laravel mereka.

## UCAPAN TERIMA KASIH

Penulis bersyukur kepada Tuhan Yang Maha Esa, Pencipta alam semesta, dan ingin mengungkapkan rasa terima kasih kepada semua individu dan kelompok yang telah memberikan bantuan, dukungan, dan arahan berharga selama pelaksanaan penelitian ini.

## REFERENSI / REFERNCES

- [1] Piumali, KGH, dan H. Budditha, "Intelligent Form Generator Using Expert Systems," *International Journal of Scientific and Research Publications*, vol. 13, no. 1, Jan. 2023, pp. 101-119, doi:10.29322/IJSRP.13.01.2023.p13313. 2023, pp. 36-48, doi:10.59603/jpmnt.v2i1.257.
- [2] D. Syarizka, "Perkembangan dan Masa Depan Industri SaaS di Indonesia," *Tech In Asia*, preprint, hlm, 100.
- [3] Ramadan, R., Muslimah, A.-Z.H. dan Rokhmawati, R.I. "Perancangan User Interface Aplikasi EzyPay menggunakan Metode Design Sprint (Studi Kasus PT. Arta Elektronik Indonesia)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 3(9), hal. 8831-8840, 2019.
- [4] T. Černý, dan M. Donahoo, *Formbuilder: A novel approach to deal with view development and maintenance*, Proceedings of Student Research Forum, Slovakia, 2011.
- [5] L. Welling, L. Thomson, *PHP and MySql Development*, 2 ed, Sams Publishing.
- [6] R.S, Pressman, dan B.R. Maxim, "Software Engineering: A Practitioner's Approach," 9 ed, McGraw-Hill Education, 2019.
- [7] I. Sommerville, *Engineering Software Products: An Introduction to Modern Software Engineering*, 1 ed. Boston: Pearson, 2008.
- [8] I. Vanderlei, J. Araujo, R. Rocha, G. Silva, F. Pacheco and J. Dantas, "Analysis of Laravel Framework Security Techniques Against Web Application Attacks," *2021 16th Iberian*

*Conference on Information Systems and Technologies (CISTI)*, Chaves, Portugal, 2021, pp. 1-7, doi: 10.23919/CISTI52073.2021.9476475.

- [9] S. M. Husain, L. Azhari, M. L. Aksani, S. A. Saputra, "Analisis Dan Implementasi Fitur Keamanan Aplikasi Pada Framework Laravel," *JIKA (Jurnal of Informatics) Universitas Muhammadiyah Tangerang*, vol. 8, no. 2, Jul. 2024, pp. 281-287 doi: 10.31000/jika.v8i3.11198.
- [10] F. Sinlea, E. Irwanda, Z. Maulana, V. E. Syahputra, "Penggunaan Framework Laravel dalam Membangun Aplikasi Website Berbasis PHP," *Jurnal Siber Multi Disiplin*, vol. 2, no. 2, Jul. 2024, pp. 119-132, doi: 10.38035/jsmd.v2i2.186.



Sigit Wijanarko lahir di Sintang pada tanggal 23 Juni 1980. Lulus S1 Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Pembangunan Nasional – Jakarta pada tahun 2004. Lulus Magister Ilmu Komputer Program Pascasarjana Universitas Budi Luhur dengan konsentrasi Rekayasa Komputasi Terapan pada tahun 2019. Saat ini aktif sebagai Dosen tetap di STMIK Antar Bangsa dan praktisi IT di perusahaan swasta.