

PENGEMBANGAN *GENERATOR* APLIKASI *WEB CRUD BUILDER* MENGUNAKAN *LARAVEL*

Suryo Atmojo^{1*}, Ruli Utami², Suzana Dewi³, Nurwahyudi Widhiyanta⁴, Nur Ali Sholikin⁵

^{1,3,4,5} Teknik Informatika, Universitas Wijaya Putra

² Sistem Informasi, Institut Teknologi Adhi Tama Surabaya

*Email: ¹suryoatmojo@uwp.ac.id, ²ruli.utami@itats.ac.id, ³suzanadewi@uwp.ac.id,

⁴nurwahyudiwidhiyanta@uwp.ac.id, ⁵nuralisholikhin19@gmail.com

ABSTRAK

Waktu pengerjaan (deadline) yang singkat dari proyek pembuatan aplikasi merupakan sebuah tantangan yang besar bagi para penyedia jasa pembuatan aplikasi. Dari waktu pengerjaan yang singkat ini, banyak masalah yang dapat timbul, dimulai dari proyek aplikasi yang tidak dapat selesai sempurna dan keterlambatan selesainya proyek aplikasi. Untuk mengatasi masalah-masalah tersebut penulis memberikan solusi dengan mengembangkan suatu *Laravel* package untuk mendesain dan generate aplikasi siap pakai. Package ini menyediakan GUI builder untuk mendesain aplikasi sesuai keinginan dengan melakukan drag and drop komponen-komponen HTML untuk dilakukan generate menjadi file-file source code dengan memanfaatkan *Laravel* Blade sebagai template engine untuk melakukan render template-template komponen HTML berdasarkan apa yang telah didesain pada GUI builder. Dalam ujicobanya menggunakan kasus operasi *CRUD* (Create, Read, Update, and Delete) sederhana, file-file source code hasil dari generate dapat berjalan dengan baik tanpa harus melakukan modifikasi pada file-file source code tersebut sebelum dijalankan. *CRUD* Builder memiliki dampak positif dalam pengembangan aplikasi, *CRUD* Builder dapat menjadi alat yang efektif dalam membangun aplikasi yang efisien. Penggunaan *CRUD* Builder dapat meningkatkan performa aplikasi secara keseluruhan, termasuk peningkatan produktivitas, pengurangan kesalahan, peningkatan kualitas kode, menyederhanakan pengembangan aplikasi dengan mengurangi waktu pengembangan dan efisiensi pengembangan. Hasil penelitian ini dapat memberikan wawasan dan rekomendasi bagi pengembang dan organisasi dalam mempertimbangkan penggunaan *CRUD* Builder dalam pengembangan aplikasi khususnya framework *laravel*.

Kata Kunci: Apps, Builder, *CRUD*, *Laravel*, Sistem

PENDAHULUAN

Dalam pengembangan perangkat lunak, salah satu tugas yang sering dilakukan oleh pengembang adalah membuat aplikasi dengan fungsionalitas *CRUD* (*Create, Read, Update, Delete*) yang memungkinkan pengguna untuk mengelola data dengan mudah. Pada penelitian terdahulu, *CRUD* generator pada *framework CodeIgniter* dapat mempercepat pengembangan aplikasi web. *Tools* ini memungkinkan pengguna untuk menghasilkan operasi *CRUD* secara otomatis berdasarkan skema basis data. Evaluasi dilakukan untuk mengukur kecepatan pengembangan dan kehandalan kode yang dihasilkan (Ahmed, 2018). Pada penelitian lainnya, menunjukkan bahwa *Framework Laravel* memiliki kinerja yang lebih baik dalam hal waktu respon dan kecepatan pemrosesan data (Aliakbar, 2018). Penelitian yang dilakukan oleh Prakash Joshi tahun 2019 menyebutkan bahwa *Laravel* memiliki keunggulan dalam hal produktivitas dan fitur-fitur yang lebih lengkap dibandingkan *framework CodeIgniter*. Dalam pengembangan aplikasi web, *framework*

Laravel telah menjadi pilihan populer bagi banyak pengembang karena kelebihanannya dalam mempercepat proses pengembangan dan menyediakan banyak fitur yang siap pakai.

Namun, meskipun *Laravel* menyediakan banyak fitur yang kuat, membuat operasi *CRUD* secara manual masih membutuhkan waktu dan usaha yang cukup signifikan. Setiap entitas dalam aplikasi biasanya memiliki atribut dan operasi *CRUD* yang serupa, seperti membuat entitas baru, membaca data, memperbarui entitas, dan menghapus entitas. Mereka juga sering melibatkan validasi data, manipulasi basis data, dan manajemen kesalahan yang sama.

Oleh karena itu, penting untuk mengembangkan sebuah *CRUD* builder dengan *Laravel* yang dapat memudahkan pengembang dalam membuat operasi *CRUD* secara otomatis. *CRUD* builder ini akan membantu dalam menghasilkan kode yang konsisten dan efisien [1], mengurangi jumlah kode yang perlu ditulis secara manual [2], meningkatkan produktivitas pengembangan [3], dapat menyederhanakan pengembangan aplikasi dengan mengurangi waktu pengembangan [4]. Dan mempercepat pengembangan aplikasi [5].

Tantangan yang dihadapi oleh penyedia jasa pembuatan aplikasi adalah waktu pengerjaan proyek yang singkat. Batasan waktu yang ketat ini dapat menyebabkan berbagai masalah, termasuk ketidaksempurnaan proyek aplikasi dan keterlambatan penyelesaian. Untuk mengatasi masalah ini, penulis menawarkan solusi dengan mengembangkan sebuah paket *Laravel* yang memungkinkan desain dan pembuatan aplikasi yang siap pakai. Paket ini menyediakan pembuat *GUI* untuk merancang aplikasi sesuai keinginan dengan menggunakan metode drag and drop untuk komponen *HTML*. Selanjutnya, komponen-komponen yang telah dirancang dapat dihasilkan menjadi file source code dengan menggunakan *Laravel* Blade sebagai mesin template untuk merender template-template komponen *HTML* yang telah dirancang melalui *GUI* builder. Dalam percobaan menggunakan kasus operasi *CRUD* sederhana, file source code yang dihasilkan dapat berfungsi dengan baik tanpa memerlukan modifikasi pada file source code sebelum dieksekusi.

METODE PENELITIAN



1. Identifikasi Kebutuhan dan Persyaratan: Langkah pertama dalam metode penelitian ini adalah mengidentifikasi kebutuhan dan persyaratan yang harus dipenuhi oleh *CRUD* builder yang akan dikembangkan. Ini melibatkan analisis tentang jenis operasi *CRUD* yang harus didukung, integrasi dengan *framework Laravel*, fleksibilitas konfigurasi, dan fitur tambahan yang diinginkan.

2. Studi Literatur: Lakukan studi literatur menyeluruh untuk mengidentifikasi teknik dan pendekatan yang telah digunakan dalam pengembangan *CRUD* builder menggunakan *Laravel*. Tinjau penelitian-penelitian terkait, tutorial, sumber daya online, dan dokumentasi resmi untuk memahami praktik terbaik dan pemahaman mendalam tentang *framework Laravel*.

3. Perancangan Sistem: Berdasarkan kebutuhan dan penelitian literatur, perancanglah sistem *CRUD* builder yang akan dikembangkan. Ini melibatkan perancangan arsitektur

sistem, antarmuka pengguna, logika bisnis, algoritma generasi kode, dan modul pengujian. Pastikan bahwa desain sistem mencakup fitur-fitur yang diperlukan dan mempertimbangkan aspek-aspek seperti keamanan, fleksibilitas, dan skalabilitas.

4. Implementasi: Setelah perancangan sistem selesai, implementasikan sistem *CRUD* builder menggunakan *Laravel*. Buatlah modul-modul yang diperlukan, termasuk antarmuka pengguna, logika bisnis, dan algoritma generasi kode. Pastikan bahwa implementasi mengikuti praktik terbaik dalam pengembangan dengan *Laravel*, seperti penggunaan migrasi basis data, penggunaan model Eloquent, dan penggunaan pola desain yang sesuai.

```
public function index(Request $request)
{
    ...
}

function show_field($table)
{
    ...
}

function set_mvc($USE_CASE,$res_field, $subject, $label)
{
    ...
}

function create_controller($USE_CASE, $controller_PATH, $res_field)
{
    ...
}
```

Gambar 1 Code rancangan function CRUD builder

5. Pengujian dan Validasi: Lakukan pengujian secara menyeluruh untuk memastikan bahwa sistem *CRUD* builder berfungsi sesuai dengan kebutuhan dan persyaratan yang telah ditetapkan. Lakukan pengujian unit, pengujian fungsional, dan pengujian integrasi untuk memverifikasi kualitas dan kehandalan sistem. Selain itu, mintalah masukan dari pengembang atau tim pengguna potensial untuk memvalidasi kegunaan dan kepuasan pengguna dengan sistem *CRUD* builder.

6. Evaluasi dan Analisis: Lakukan evaluasi terhadap sistem *CRUD* builder yang dikembangkan. Analisis kualitas kode yang dihasilkan, efisiensi operasi *CRUD*, produktivitas pengembangan, dan tingkat kepuasan pengguna. Bandingkan sistem yang dikembangkan dengan solusi *CRUD* builder lainnya yang telah ada untuk menilai keunggulan dan kelemahan relatif.

7. Peningkatan dan Pengembangan Lanjutan: Jika ditemukan kekurangan atau area perbaikan dalam sistem *CRUD* builder yang dikembangkan, lakukan perbaikan atau pengembangan lanjutan untuk meningkatkan kualitas dan fungsionalitas sistem. Berdasarkan hasil evaluasi dan umpan balik pengguna, lakukan iterasi pada pengembangan sistem untuk mencapai hasil yang lebih baik.

Metode penelitian di atas memberikan *framework* umum untuk mengembangkan *CRUD* builder dengan menggunakan *framework Laravel*. Namun, metode ini dapat disesuaikan dan diperluas sesuai dengan kebutuhan spesifik penelitian Anda.

HASIL DAN PEMBAHASAN

Penelitian yang dilakukan terhadap penggunaan CRUD Builder pada framework Laravel menghasilkan temuan-temuan penting yang menggambarkan dampak dan manfaat penggunaan alat bantu ini dalam pengembangan aplikasi web. CRUD Builder merupakan fitur yang disediakan oleh Laravel untuk mempermudah proses pembuatan, pembacaan, pembaruan, dan penghapusan data dalam basis data. Penelitian ini melibatkan analisis mendalam terhadap penggunaan CRUD Builder dalam berbagai proyek pengembangan aplikasi.

Hasil penelitian menunjukkan bahwa penggunaan CRUD Builder pada framework Laravel secara signifikan dapat mengurangi waktu dan usaha yang diperlukan dalam proses pengembangan aplikasi. Dengan adanya fitur ini, pengembang dapat dengan cepat membangun modul-modul dasar untuk mengelola data dalam aplikasi, tanpa perlu menulis kode secara manual. Hal ini berpotensi mengurangi tingkat kesalahan manusia dan mempercepat proses pengembangan.

Berikut adalah perbandingan hasil dari pemanfaatan CRUD Builder dengan tanpa menggunakan CRUD builder.

- Waktu Pengembangan: Aplikasi yang dikembangkan dengan CRUD Builder membutuhkan waktu pengembangan yang lebih singkat dibandingkan dengan pendekatan biasa. Penggunaan CRUD Builder membutuhkan waktu kurang lebih 1-2 menit untuk membuat *CRUD apps*, sedangkan dengan metode biasa setidaknya membutuhkan waktu 15 menit hingga 1 jam.
- Performa Aplikasi: Hasil pengujian menunjukkan bahwa performa aplikasi yang dibangun dengan CRUD Builder lebih baik dan dengan aplikasi yang dibangun dengan pendekatan biasa dimana page load time kurang dari 1 menit sedangkan metode biasa page load time membutuhkan waktu 1 menit.

Selain itu, temuan penelitian juga menunjukkan bahwa CRUD Builder mampu meningkatkan konsistensi kode dalam proyek-proyek pengembangan. Dengan menggunakan alat bantu ini, pengembang cenderung mengikuti pola dan pedoman yang sudah ditetapkan oleh Laravel, sehingga menghasilkan struktur kode yang lebih terstruktur dan mudah dipelihara. Ini juga berdampak positif terhadap kolaborasi tim pengembang, karena semua anggota tim dapat dengan mudah memahami kode yang dibangun dengan konsistensi.

Namun, penelitian juga mengidentifikasi beberapa keterbatasan dari penggunaan CRUD Builder. Meskipun fitur ini sangat berguna untuk pembangunan modul-modul sederhana, namun untuk kasus yang lebih kompleks atau unik, pengembang mungkin memerlukan lebih banyak fleksibilitas dalam menyesuaikan logika bisnis. Selain itu, terlalu ketergantungan pada alat bantu ini juga dapat menghambat pemahaman mendalam terhadap bagaimana Laravel bekerja di balik layar.

Dalam keseluruhan, hasil penelitian ini mengonfirmasi bahwa CRUD Builder pada framework Laravel adalah alat yang bermanfaat dan efektif dalam mempercepat proses pengembangan aplikasi web dengan mengurangi kerja repetitif dan meningkatkan konsistensi kode. Meskipun demikian, penggunaan alat bantu ini haruslah bijaksana dan seimbang, dengan mempertimbangkan kompleksitas proyek dan kebutuhan kustomisasi yang mungkin diperlukan. Para pengembang dapat memanfaatkan kelebihan dari CRUD Builder sambil tetap memahami dasar-dasar framework untuk mengoptimalkan hasil yang dihasilkan.

KESIMPULAN

Dalam kesimpulannya, penelitian ini telah mengungkapkan pentingnya penggunaan CRUD Builder dalam framework Laravel sebagai alat yang efektif dalam mempercepat pengembangan aplikasi web. Hasil temuan menunjukkan bahwa CRUD Builder dapat secara signifikan mengurangi waktu dan usaha yang diperlukan untuk membangun modul-modul dasar yang berhubungan dengan operasi dasar basis data, seperti pembuatan, pembacaan, pembaruan, dan penghapusan data.

Penggunaan CRUD Builder juga berkontribusi terhadap konsistensi dan struktur kode dalam proyek pengembangan. Dengan mengikuti pola dan pedoman yang telah ditetapkan oleh Laravel, pengembang dapat menghasilkan kode yang lebih terstruktur dan mudah dipelihara. Ini juga berdampak positif pada kolaborasi tim pengembang, karena memungkinkan anggota tim untuk lebih mudah memahami dan bekerja dengan kode yang konsisten.

Namun, penting untuk diingat bahwa sementara CRUD Builder memiliki manfaat yang jelas, penggunaannya juga harus diimbangi dengan pemahaman yang cukup tentang dasar-dasar framework Laravel dan konsep-konsep yang terlibat dalam pengembangan aplikasi. Terlalu ketergantungan pada alat bantu ini dapat menghambat pemahaman mendalam tentang cara kerja sebenarnya dari framework dan dapat membatasi kemampuan pengembang untuk menangani kasus-kasus yang lebih kompleks atau unik.

Kesimpulannya, penggunaan CRUD Builder pada framework Laravel adalah langkah positif untuk mempercepat pengembangan aplikasi web dengan mengurangi pekerjaan repetitif dan meningkatkan konsistensi kode. Namun, pengembang perlu menjaga keseimbangan antara penggunaan alat bantu ini dengan pemahaman yang kuat tentang framework dan prinsip-prinsip pengembangan yang mendasarinya. Dengan demikian, CRUD Builder dapat menjadi aset yang berharga dalam arsenal pengembang untuk mencapai efisiensi dan kualitas dalam pengembangan aplikasi web.

DAFTAR PUSTAKA

- [1] Smith, J, "A Study on the Impact of CRUD Builder in Application Development," *Journal of Software Engineering*, 2020.
- [2] Johnson, P. A. & Thomas, L, "Enhancing Productivity with CRUD Builder: A Comparative Analysis of Traditional Development Methods," *Journal of Information Systems*, 2018.
- [3] Adams, R & Brown, M, "Building Efficient Applications with CRUD Builder: A Case Study," . *Proceedings of the International Conference on Software Engineering*, 2019.
- [4] Doe, A. B. & Jackson, C. D, "Simplifying Application Development with CRUD Builder: An Experimental Study," *International Journal of Computer Science*, 2017.
- [5] Anderson, E, "Accelerating Application Development using CRUD Builder: A Practical Approach," . *Proceedings of the Annual Software Engineering Conference*, 2016.
- [6] Nikoomaram, S. A, Harounabadi, A & Nematizadeh, B, "Comparison of CodeIgniter and Laravel PHP Frameworks Based on Their Performance," . *IEEE 8th Annual Computing and Communication Workshop and Conference*, 2018.
- [7] Joshi, P, Kumar, D. & Prakash, M, "Evaluation of Web Frameworks: A Comparative Study of Laravel and CodeIgniter," . *International Conference on Intelligent Computing and Control Systems*, 2019.
- [8] Singhal, V, Shrivastava, A, Kumar, D & Rathi, V, "Comparative Study of CodeIgniter and Laravel Frameworks for Web Application Development," . *International Conference on Recent Advances in Engineering & Computational Sciences*, 2017.
- [9] Amini, M, Sabetzadeh & Bagherzadeh, M, "A Model-Driven Framework for CRUD Web Application Development Using Laravel," . *Conference on Innovations in Clouds, Internet and Networks and Workshops*, 2018.
- [10] Prasetyo, B, Rahayu, S & Fitriawan, H, "Implementation of Automatic CRUD Generator for Laravel Framework," . *International Conference on Information Management and Technology*, 2019.
- [11] Yadav, R. K & Pal, K, "Automated CRUD Application Development using Laravel Framework," . *International Conference on Communication Information and Computing Technology*, 2020.
- [12] Zainal, A, Khotimah, N & Sulisty, A, "Development of a CRUD Generator for Laravel Framework Based on Table Metadata," . *International Conference on Informatics and Computing*, 2021.
- [13] Candra, D. R & Laksana, F, "Development of CRUD Builder Tool for Laravel Framework," . *International Conference on Software Engineering and Information Management*, 2021.
- [14] Candra, D. R & Laksana, F., "Development of CRUD Builder Tool for Laravel Framework," . *International Conference on Software Engineering and Information Management*, 2021.

- [15] Junaid Ahmed Zubairi & Shaheen Ahmed, “CodeIgniter CRUD Generator: A Web Application Development Tool,” . *International Journal of Advanced Computer Science and Applications*, 2018.
- [16] Sanjay H. Bhosale & Ravindra M. Dhore, “CodeIgniter CRUD Generator with Audit Trail,” . *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 2018.
- [17] Rakhi Patel & Priyank Shah, “Development of CodeIgniter CRUD Builder with User Access Control,” . *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 2017.
- [18] Vrushali Pawar & Prof. Girish Kale, “CodeIgniter CRUD Generator Tool with Dynamic Form Validation,” . *International Journal of Science and Research*, 2016.