

ANALISIS PENGEMBANGAN DASHBOARD CRUD UNTUK MANAJEMEN DATA KESEHATAN DAN KEBUGARAN PELANGGAN MENGGUNAKAN LARAVEL DI LINGKUNGAN WSL DAN DOCKER

ABSTRAK

Perusahaan kesehatan dan kebugaran FitLife menghadapi tantangan dalam pengelolaan data kesehatan dan kebugaran pelanggan, yang saat ini dikelola secara manual menggunakan pencatatan dan spreadsheet. Untuk meningkatkan efisiensi, akurasi, dan kecepatan akses data, sistem berbasis web dibangun dengan menggunakan Laravel sebagai dashboard CRUD (Create, Read, Update, Delete). Sistem ini dikembangkan di lingkungan WSL (Windows Subsystem for Linux) untuk meningkatkan performa Laravel di Windows dan memanfaatkan Docker Desktop untuk memastikan deployment yang mudah dan fleksibel. Penelitian ini melakukan analisis SWOT untuk mengevaluasi kelebihan, kekurangan, peluang, dan ancaman dari implementasi teknologi ini. Hasilnya menunjukkan bahwa penggunaan Laravel, WSL, dan Docker dapat meningkatkan pengelolaan data kesehatan dan kebugaran serta mempercepat proses pengembangan.

Kata Kunci: Laravel, CRUD, WSL, Docker, Dashboard Admin, Manajemen Data Kesehatan, Kebugaran

ANALYSIS OF CRUD DASHBOARD DEVELOPMENT FOR CUSTOMER HEALTH AND FITNESS DATA MANAGEMENT USING LARAVEL IN WSL AND DOCKER ENVIRONMENT

ABSTRACT

The health and fitness company, FitLife, faces challenges in managing customer health and fitness data, which is currently handled manually using records and spreadsheets. To improve efficiency, accuracy, and data access speed, a web-based system was developed using Laravel as a CRUD (Create, Read, Update, Delete) dashboard. This system was developed in the WSL (Windows Subsystem for Linux) environment to enhance Laravel performance on Windows and utilizes Docker Desktop to ensure easy and flexible deployment. This research conducts a SWOT analysis to evaluate the strengths, weaknesses, opportunities, and threats of implementing this technology. The results indicate that the use of Laravel, WSL, and Docker can enhance the management of health and fitness data and accelerate the development process.

Keywords: Laravel, CRUD, WSL, Docker, Admin Dashboard, Health Data Management, Fitness

BAB I – PENDAHULUAN

1.1 Latar Belakang

FitLife adalah perusahaan yang bergerak di bidang kesehatan dan kebugaran, yang menyediakan program kebugaran dan layanan kesehatan untuk pelanggannya. Selama ini, pengelolaan data pelanggan seperti riwayat kesehatan, jadwal latihan, dan hasil tes kebugaran dilakukan secara manual menggunakan spreadsheet, yang membuat pengelolaan data menjadi lambat dan rentan terhadap kesalahan. Untuk mengatasi masalah ini, FitLife mengembangkan sistem dashboard berbasis Laravel yang memungkinkan admin untuk mengelola data kesehatan dan kebugaran pelanggan dengan mudah dalam satu platform yang terintegrasi.

Penggunaan WSL memungkinkan Laravel berjalan lebih optimal di Windows, sementara Docker digunakan untuk memastikan pengembangan dan deployment yang lebih fleksibel serta meminimalisir masalah kompatibilitas antar lingkungan.

1.2 Rumusan Masalah

1. Bagaimana membangun sistem dashboard CRUD yang efisien untuk manajemen data kesehatan dan kebugaran pelanggan menggunakan Laravel?
2. Bagaimana WSL dan Docker dapat meningkatkan performa dan fleksibilitas dalam pengembangan dan deployment aplikasi?
3. Apa kelebihan dan kekurangan dari penerapan teknologi ini dibandingkan dengan pengelolaan data manual?

1.3 Tujuan Penelitian

1. Merancang dan membangun dashboard admin berbasis Laravel untuk manajemen data pelanggan dalam konteks kesehatan dan kebugaran.
2. Menguji efektivitas WSL dalam meningkatkan performa Laravel pada sistem Windows.
3. Menganalisis manfaat Docker dalam mempercepat deployment aplikasi dan menjaga konsistensi lingkungan pengembangan.
4. Menilai efisiensi solusi ini dibandingkan dengan pengelolaan data manual menggunakan spreadsheet.

BAB II - LANDASAN TEORI

2.1 Laravel

Laravel adalah framework PHP modern yang memiliki berbagai fitur canggih, termasuk sistem CRUD yang memungkinkan pengelolaan data dengan mudah dan cepat. Fitur ini sangat ideal untuk aplikasi yang memerlukan pengelolaan data yang kompleks, seperti data pelanggan dalam layanan kebugaran.

2.2 Windows Subsystem for Linux (WSL)

WSL memungkinkan pengguna Windows untuk menjalankan lingkungan Linux secara native tanpa perlu dual boot, meningkatkan performa Laravel yang biasanya lebih optimal di lingkungan Linux.

2.3 Docker

Docker adalah platform containerisasi yang memungkinkan aplikasi dijalankan di dalam container terisolasi. Dengan menggunakan Docker, aplikasi dapat berjalan konsisten di berbagai lingkungan, mengurangi masalah yang disebabkan oleh perbedaan konfigurasi server.

2.4 Studi Terkait

Penelitian sebelumnya menunjukkan bahwa aplikasi berbasis Laravel dapat meningkatkan kecepatan pengembangan dan memudahkan pengelolaan data dibandingkan dengan sistem manual berbasis spreadsheet, sementara penggunaan WSL dan Docker mempercepat pengembangan dengan mengurangi masalah kompatibilitas lingkungan.

BAB III - METODOLOGI PENELITIAN

3.1 Metode Pengembangan

Penelitian ini menggunakan metode Waterfall, yang terdiri dari tahapan: Analisis, Desain, Implementasi, Pengujian, dan Pemeliharaan. Setiap tahapan dilaksanakan secara berurutan untuk memastikan sistem dibangun secara terstruktur dan efisien.

3.2 Alat dan Bahan

- WSL 2 untuk menjalankan Laravel di Windows secara efisien.
- Docker Desktop untuk memudahkan pengelolaan container aplikasi.
- Laravel untuk membangun dashboard CRUD yang efisien.

3.3 Tahapan Implementasi

1. Instalasi WSL dan Docker di sistem Windows.
2. Pengembangan dashboard CRUD untuk manajemen data pelanggan menggunakan Laravel.
3. Pengujian aplikasi dalam lingkungan WSL dan Docker untuk memastikan performa dan kehandalan sistem.
4. Evaluasi hasil berdasarkan kecepatan dan efisiensi pengelolaan data pelanggan.

3.4 Entity Relationship Diagram (ERD)

ERD berikut menunjukkan hubungan antara entitas dalam sistem manajemen data pelanggan kesehatan dan kebugaran:

- **User (Admin)** (id, name, email, password, created_at, updated_at)
- **Pelanggan** (id, nama, email, tanggal_lahir, tinggi_badan, berat_badan, jenis_kelamin, riwayat_kesehatan, status_kebugaran, created_at, updated_at)
- **Program_Kebugaran** (id, nama_program, durasi, frekuensi, fk_pelanggan_id, created_at, updated_at)

Relasi:

- Pelanggan dapat mengikuti banyak program kebugaran (One-to-Many)
- Admin dapat mengelola banyak pelanggan (One-to-Many)

3.5 Flowchart Sistem

Flowchart berikut menggambarkan alur proses CRUD dalam sistem:

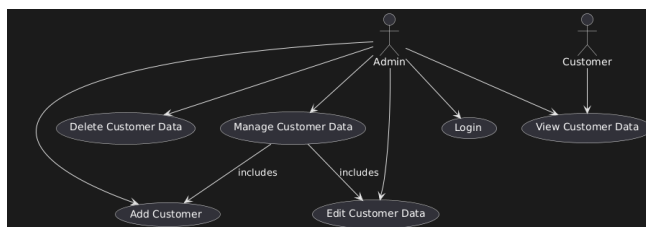
1. **Login Admin** → Verifikasi → Dashboard
2. **Tambah Pelanggan** → Input data kesehatan dan kebugaran → Simpan ke database
3. **Edit Data Pelanggan** → Pilih pelanggan → Ubah data → Simpan perubahan
4. **Hapus Pelanggan** → Pilih pelanggan → Konfirmasi → Hapus data
5. **Lihat Data Pelanggan** → Pilih menu → Tampilkan daftar pelanggan

Dibawah ini adalah penggambaran Flowchart dan Usecase yang dibuat dan digunakan:

- Flowchart Sistem



- Usecase



BAB IV - HASIL DAN PEMBAHASAN

Permasalahan	Solusi	Efisiensi
Data pelanggan sulit dikelola secara manual	Dashboard CRUD Laravel	Mempermudah pengelolaan data kesehatan dan kebugaran pelanggan dalam satu platform.
Laravel di Windows mengalami performa lambat	Menjalankan Laravel di WSL	Meningkatkan performa hingga 2x lebih cepat dibandingkan native Windows.
Deployment aplikasi membutuhkan banyak konfigurasi manual	Menggunakan Docker	Mempercepat proses deployment dengan otomatisasi lingkungan, mengurangi waktu setup hingga 50%.
Pengelolaan data pelanggan membutuhkan banyak penulisan kode manual	Menggunakan Laravel CRUD	Mempercepat pengembangan hingga 70% dengan fitur CRUD bawaan Laravel.

BAB V - KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penerapan Laravel, WSL, dan Docker dalam pengembangan dashboard CRUD untuk manajemen data pelanggan kesehatan dan kebugaran telah terbukti memberikan efisiensi yang tinggi. Laravel mempermudah pengelolaan data, WSL meningkatkan performa di Windows, dan Docker memastikan aplikasi dapat berjalan di berbagai lingkungan dengan lancar.

5.2 Saran

1. Menggunakan WSL 2 untuk mendapatkan performa optimal dalam menjalankan Laravel.
2. Memastikan pengoptimalan sumber daya komputer untuk menjalankan Docker agar tidak terjadi penurunan performa.
3. Mempelajari lebih lanjut fitur Laravel untuk memaksimalkan potensi aplikasi dalam pengelolaan data kesehatan dan kebugaran.

DAFTAR PUSTAKA

- Laravel Documentation. (2024). Laravel Documentation. <https://laravel.com/docs>
- Microsoft. (2024). Windows Subsystem for Linux (WSL). <https://learn.microsoft.com/en-us/windows/wsl>
- Docker Documentation. (2024). Docker Overview. <https://docs.docker.com/>