



JEMSI:
**Jurnal Ekonomi Manajemen Sistem
Informasi**

E-ISSN: 2686-5238
P-ISSN: 2686-4916

<https://dinastirev.org/JEMSI> dinasti.info@gmail.com +62 811 7404 455

DOI: <https://doi.org/10.38035/jemsi.v6i5>
<https://creativecommons.org/licenses/by/4.0/>

Implementasi Arsitektur Microservice pada Sistem Informasi HRD Berbasis WEB

Ibrahim Cahya Gumilar¹, Alun Sujjada², Kamdan Kamdan³

¹Universitas Nusa Putra, Sukabumi, Indonesia, ibrahim.cahya_ti20@nusaputra.ac.id

²Universitas Nusa Putra, Sukabumi, Indonesia, alun.sujjada@nusaputra.ac.id

³Universitas Nusa Putra, Sukabumi, Indonesia, kamdan@nusaputra.ac.id

Corresponding Author: ibrahim.cahya_ti20@nusaputra.ac.id¹

Abstract: PT Cosmo Technology Indonesia is a technology company that manufactures Light Emitting Diodes (LEDs), with its primary markets in America and Asia. The company's recruitment process, which is still conducted manually, poses several challenges, such as the accumulation of physical documents and long processing times. This study aims to develop a web-based HRD information system using a microservice architecture to improve the efficiency and effectiveness of the recruitment process. The microservice architecture was chosen for its ability to enhance scalability, flexibility, and reliability. The system is built using the Express.js and Laravel frameworks for the backend, HTML and Bootstrap for the frontend, and MySQL as the database. The development method used is Rapid Application Development (RAD) to accelerate the system creation process. Black box testing results show that 100% of the system functions run as required, while a survey using the Likert scale received positive responses regarding improvements in recruitment efficiency and effectiveness. With this system, the company can manage its recruitment process more efficiently, reduce costs and processing time, and provide a better experience for applicants.

Keyword: HRD Information System, Employee Recruitment, Microservice Architecture, Express.js, Laravel, MySQL

Abstrak: PT Cosmo Technology Indonesia adalah perusahaan teknologi yang memproduksi Light Emitting Diode (LED) dengan pasar utama di Amerika dan Asia. Proses rekrutmen karyawan yang masih dilakukan secara manual menimbulkan berbagai kendala, seperti penumpukan dokumen fisik, dan waktu pemrosesan yang lama. Penelitian ini bertujuan mengembangkan sistem informasi HRD berbasis web dengan arsitektur *microservice* untuk meningkatkan efisiensi dan efektivitas proses rekrutmen. Arsitektur *microservice* dipilih karena kemampuannya dalam meningkatkan skalabilitas, fleksibilitas, dan reliabilitas. Sistem ini dibangun menggunakan *framework Express.js* dan *Laravel* untuk *backend*, *HTML* dan *Bootstrap* untuk *frontend*, serta *database MySQL*. Metode pengembangan yang digunakan adalah *Rapid Application Development (RAD)* untuk mempercepat proses pembuatan sistem. Hasil pengujian *black box* menunjukkan bahwa 100% fungsi sistem berjalan sesuai kebutuhan, sementara survei menggunakan skala *Likert* mendapatkan respon positif terhadap peningkatan efisiensi dan efektivitas proses rekrutmen. Dengan sistem ini, perusahaan dapat mengelola

proses rekrutmen dengan lebih efisien, mengurangi biaya dan waktu, serta memberikan pengalaman yang lebih baik bagi pelamar.

Kata Kunci: Sistem Informasi HRD, Rekrutmen Karyawan, Arsitektur Microservice, Express.js, Laravel, MySQL

PENDAHULUAN

PT Cosmo Technology Indonesia merupakan perusahaan besar di bidang teknologi yang didirikan pada tahun 2007 di Kabupaten Sukabumi. Perusahaan ini memproduksi *Light Emitting Diode (LED)* berupa lampu natal dan aksesoris pohon natal, dengan pasar utama di Amerika dan Asia. Dengan penjualan mencapai 350.000 unit setiap bulan, perusahaan menghadapi tantangan dalam proses rekrutmen karyawan (Cosmo, n.d.). Saat ini, proses rekrutmen masih dilakukan secara manual, yaitu pelamar menyerahkan lamaran dalam bentuk fisik yang kemudian diproses oleh manajemen perusahaan. Metode ini dianggap kurang efisien karena menimbulkan berbagai kendala, seperti penumpukan dokumen fisik, waktu pemrosesan yang lama, serta biaya yang harus dikeluarkan oleh pelamar saat mengirim dokumen fisik. Berdasarkan permasalahan tersebut, perusahaan memutuskan untuk mengembangkan sistem informasi HRD berbasis web dengan menggunakan arsitektur *microservice* untuk meningkatkan efisiensi dan efektivitas proses rekrutmen.

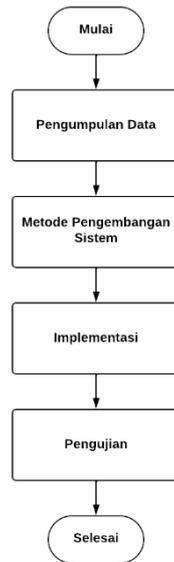
Arsitektur *microservice* adalah model arsitektur perangkat lunak di mana aplikasi dibagi menjadi layanan-layanan kecil yang independen, sehingga setiap layanan dapat dikembangkan, diuji, dan *deploy* secara terpisah tanpa mengganggu keseluruhan sistem (Risnanto, 2022). Arsitektur ini sangat cocok untuk aplikasi berbasis web yang memerlukan skalabilitas, fleksibilitas, dan kemudahan pengelolaan. Setiap layanan dapat diskalakan secara independen sesuai kebutuhan, seperti meningkatkan kapasitas layanan manajemen data pelamar tanpa memengaruhi layanan lain. Fleksibilitasnya memungkinkan penggunaan bahasa pemrograman dan *database* yang berbeda untuk setiap *service*, sehingga pengembangan dapat dilakukan secara bersamaan tanpa saling bergantung. Dalam aspek reliabilitas, kegagalan pada satu *service* tidak akan mengganggu layanan lainnya, seperti jika layanan wawancara bermasalah, layanan lain tetap dapat berfungsi. Dengan keunggulan-keunggulan tersebut, arsitektur *microservice* dipilih untuk mendukung perkembangan sistem perusahaan di masa depan.

Penelitian terkait yang telah dilakukan oleh beberapa peneliti sebelumnya menunjukkan keberhasilan penerapan arsitektur *microservice* dalam berbagai konteks. Misalnya, penelitian tentang perancangan sistem informasi desa berbasis arsitektur *microservice* (Atmojo et al., 2022) dan *reengineering* arsitektur *monolitik* ke *microservice* pada website manajemen konten (Putra Permana et al., 2023) telah membuktikan bahwa *microservice* dapat meningkatkan kinerja sistem dan memudahkan pengelolaan.

Tujuan dari penelitian ini adalah untuk mengembangkan sistem informasi HRD berbasis web dengan menggunakan arsitektur *microservice*. Sistem ini diharapkan dapat menyelesaikan berbagai masalah yang ada dalam sistem manual, seperti penumpukan dokumen fisik, dan waktu pemrosesan yang lama.

METODE

Sebelum memulai proses penelitian, perlu dilakukan perancangan alur terlebih dahulu untuk memastikan bahwa setiap langkah dalam pengembangan aplikasi berjalan secara sistematis dan terstruktur. Berikut adalah gambar 1 Tahapan Penelitian:



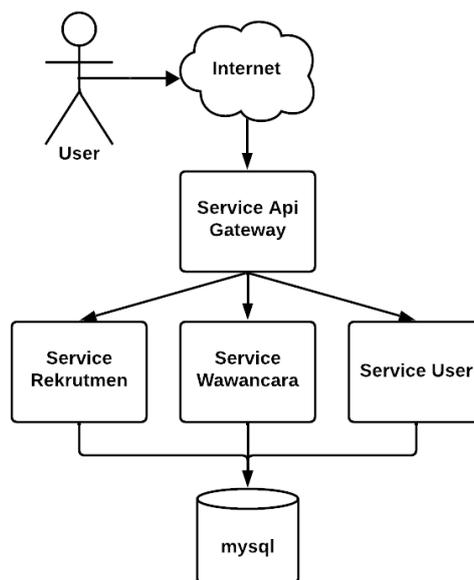
Gambar 1. Tahapan Penelitian

Pengumpulan data dilakukan dengan menggunakan metode kualitatif, di antaranya observasi, wawancara, dan studi literatur. Melalui metode ini, peneliti dapat memperoleh informasi mendalam tentang objek yang diteliti, yang kemudian dapat dimanfaatkan untuk merumuskan tujuan penelitian secara lebih akurat.

Penelitian ini menggunakan metode *Rapid Application Development* (RAD), yang memungkinkan proses pengembangan aplikasi dilakukan dalam waktu singkat (Purwati et al., 2023). Metode ini mencakup beberapa tahapan, yaitu *requirement planning* yang bertujuan mengidentifikasi permasalahan utama, *system design* untuk merancang solusi sistem, dan *implementation* sebagai tahap penerapan pengkodean berdasarkan desain yang telah dibuat (Panja & Manongga, 2023).

HASIL DAN PEMBAHASAN

Struktur Sistem



Gambar 2. Struktur Sistem

Pada gambar 2 ditunjukkan struktur sistem yang digunakan, di mana setiap service dalam sistem menggunakan port yang berbeda untuk berkomunikasi. Berikut adalah *service* yang digunakan:

a. Service Front-End (Vanilla JavaScript, Port 5500)

Service Front-End digunakan untuk menampilkan antarmuka website dan menghubungkannya dengan *back-end*. Pada penelitian ini, antarmuka dirancang menggunakan *HTML* dengan bantuan *Bootstrap* untuk memastikan tata letak yang responsif dan mudah digunakan. Pendekatan ini dipilih karena fitur yang dibutuhkan oleh sistem relatif sederhana, sehingga tidak memerlukan framework *front-end* tambahan.

b. Service API Gateway (Express.js, Port 4001)

Service API Gateway berfungsi sebagai penghubung antara berbagai *service* dalam sistem. *API Gateway* menerima permintaan dari *front-end* dan meneruskannya ke *service* lain yang relevan sesuai kebutuhan.

c. Service User (Express.js, Port 5000)

Service User bertugas menyimpan dan mengelola data akun pengguna, termasuk informasi profil, autentikasi, dan pembaruan data pengguna.

d. Service Rekrutmen (Express.js, Port 3000)

Service Rekrutmen digunakan untuk mengelola data terkait proses rekrutmen, seperti data pelamar dan informasi lowongan pekerjaan yang tersedia dalam sistem.

e. Service Wawancara (Laravel, Port 8000)

Service Wawancara berfungsi untuk menyimpan dan mengelola data terkait proses wawancara, baik untuk akun admin maupun pelamar.

Implementasi Antarmuka Sistem

Berikut adalah implementasi antarmuka sistem yang telah dilakukan berdasarkan metodologi penelitian:

1. Halaman Home

Pada halaman *home*, sebelum melakukan *login*, pengguna dapat melihat daftar lowongan pekerjaan yang tersedia. Selain itu, terdapat menu *login* yang akan mengarahkan pengguna ke halaman *login* untuk masuk ke dalam sistem. Berikut adalah gambar 3 Halaman Home:

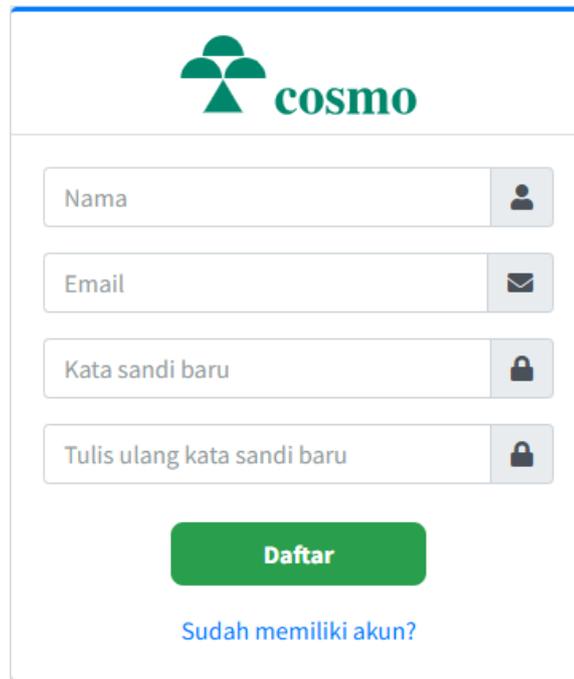


Gambar 3. Halaman Home

2. Halaman Registrasi

Pada halaman registrasi, pengguna diminta untuk memasukkan nama, email, kata sandi, dan konfirmasi kata sandi untuk mendaftarkan akun. Halaman registrasi ini menggunakan

service user yang dikembangkan dengan *framework Express.js*. Berikut adalah gambar 4 Halaman Registrasi:

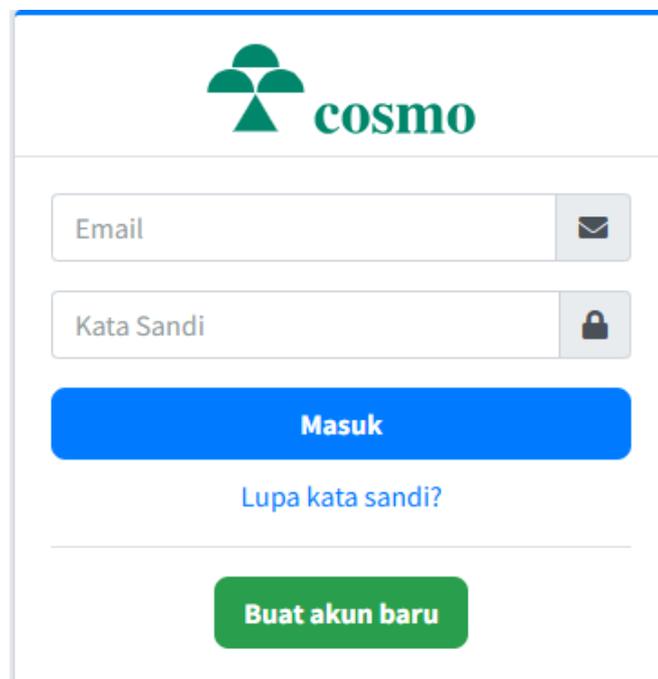


The registration page features the 'cosmo' logo at the top. Below the logo are four input fields: 'Nama' (with a person icon), 'Email' (with an envelope icon), 'Kata sandi baru' (with a lock icon), and 'Tulis ulang kata sandi baru' (with a lock icon). A green 'Daftar' button is positioned below the fields, followed by a blue link 'Sudah memiliki akun?'.

Gambar 4. Halaman Registrasi

3. Halaman *Login*

Pada halaman *login*, pengguna diminta untuk memasukkan akun yang telah dibuat pada halaman registrasi. Halaman *login* ini menggunakan *service user* yang dikembangkan dengan *framework Express.js*. Berikut adalah gambar 5 Halaman Login:

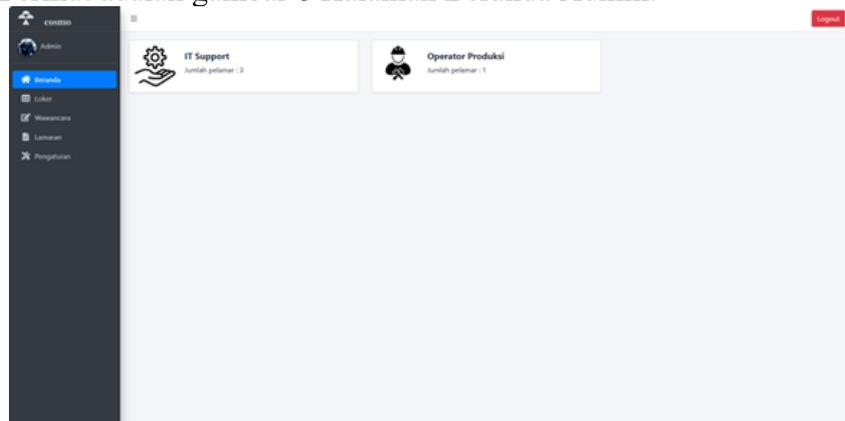


The login page features the 'cosmo' logo at the top. Below the logo are two input fields: 'Email' (with an envelope icon) and 'Kata Sandi' (with a lock icon). A blue 'Masuk' button is positioned below the fields, followed by a blue link 'Lupa kata sandi?'. A green 'Buat akun baru' button is located at the bottom of the page.

Gambar 5. Halaman Login

4. Halaman Beranda Admin

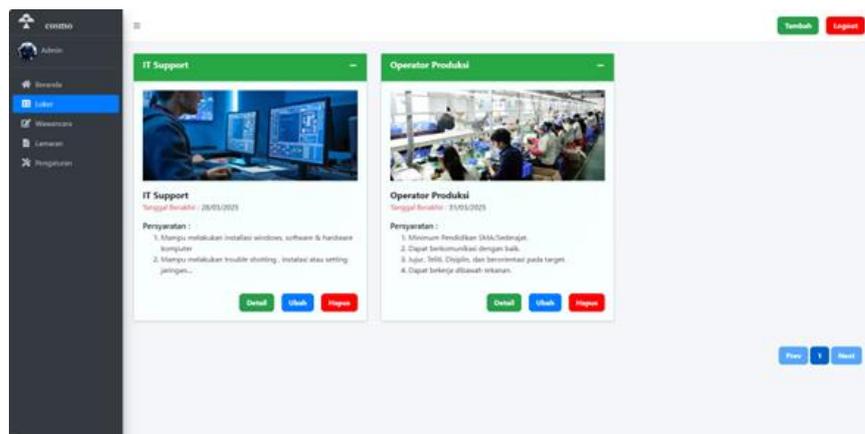
Pada halaman beranda admin, terdapat fitur untuk melihat jumlah pelamar yang sudah mendaftar. Halaman beranda admin menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 6 Halaman Beranda Admin:



Gambar 6. Halaman Beranda Admin

5. Halaman Loker Admin

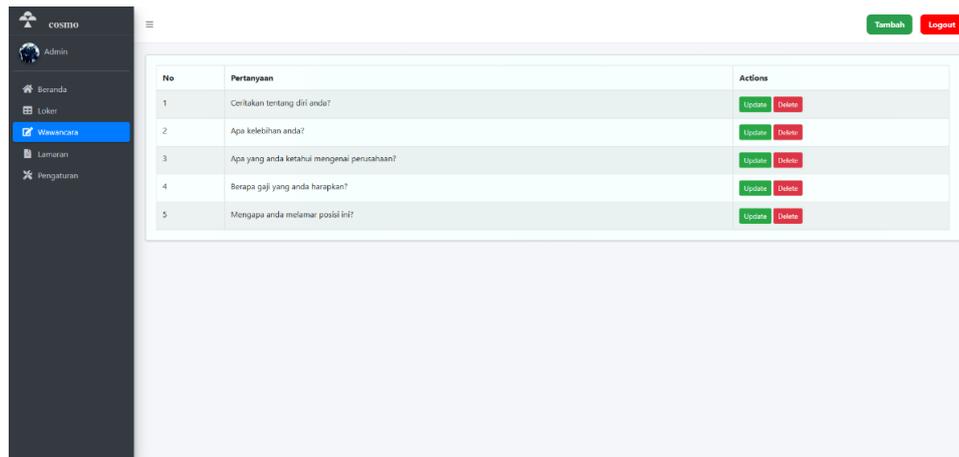
Pada halaman loker, admin dapat menambah, mengubah, menghapus, dan melihat detail lowongan pekerjaan yang sudah ditambahkan. Halaman loker admin menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 7 Halaman Loker Admin:



Gambar 7. Halaman Loker Admin

6. Halaman Wawancara Admin

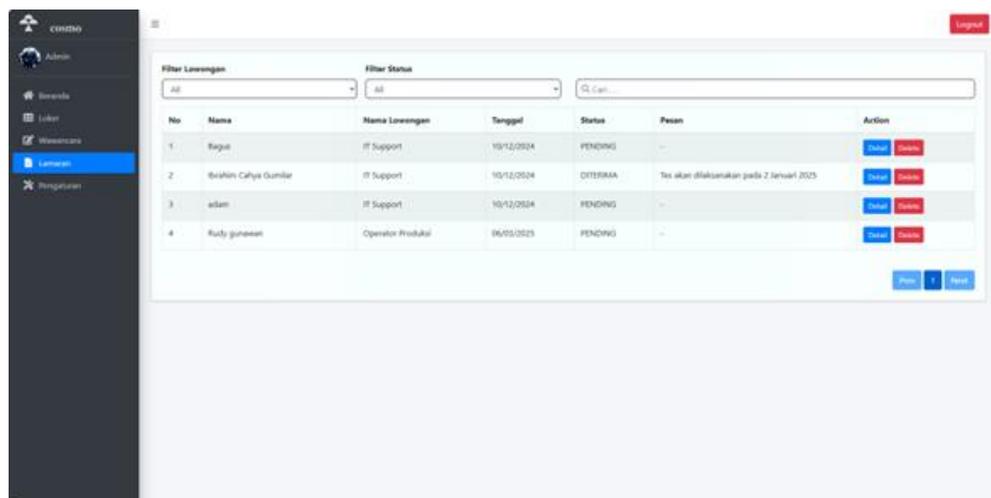
Pada halaman wawancara, admin dapat menambah, mengubah, dan menghapus pertanyaan wawancara yang nantinya akan ditampilkan pada akun pelamar. Halaman wawancara admin menggunakan *service* wawancara dengan *framework Laravel*. Berikut adalah gambar 8 Halaman Wawancara Admin:



Gambar 8. Halaman Wawancara Admin

7. Halaman Lamaran Admin

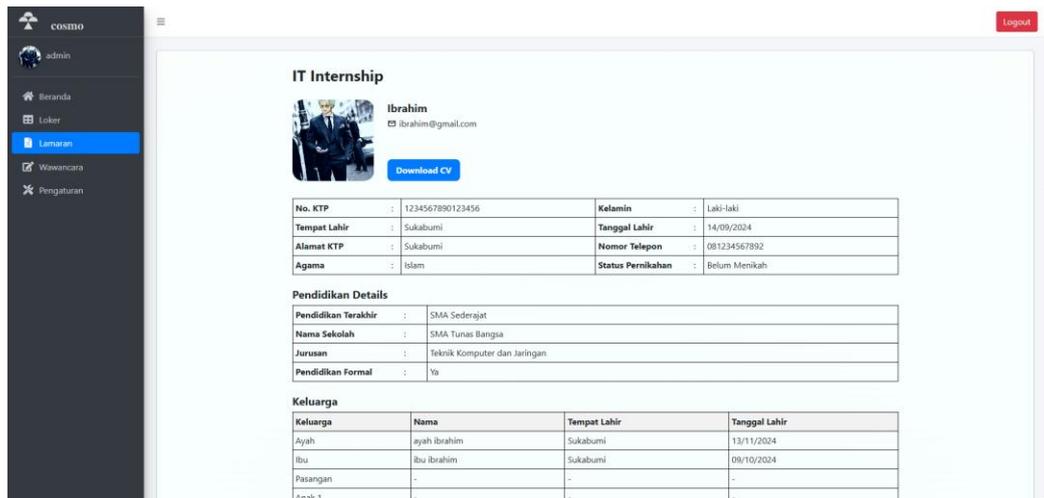
Pada halaman lamaran, admin dapat menghapus, melihat detail lamaran dengan mengklik tombol detail, mencari pelamar berdasarkan nama dan melakukan filter berdasarkan nama lowongan serta status lamaran. Halaman lamaran admin menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 9 Halaman Lamaran Admin:



Gambar 9. Halaman Lamaran Admin

8. Halaman Detail Lamaran Admin

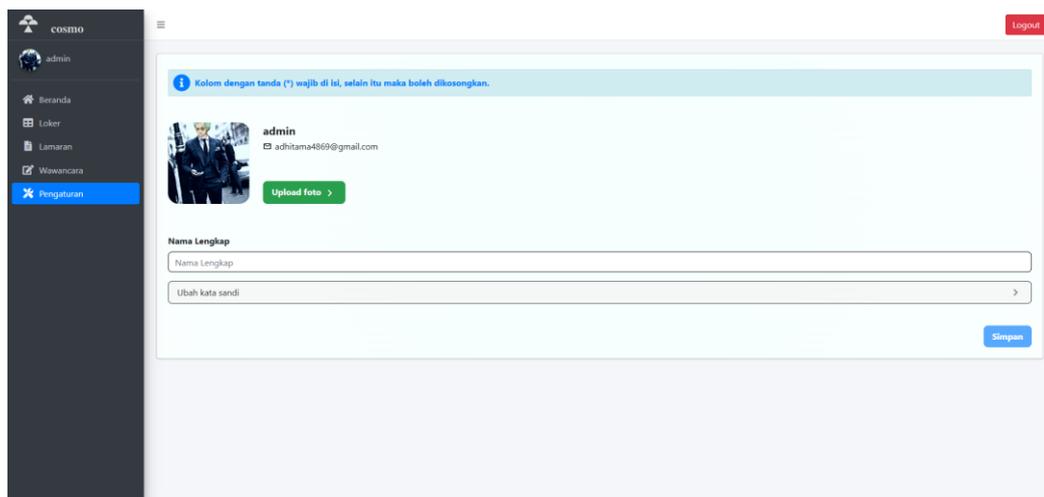
Pada halaman detail lamaran, admin dapat melihat data pelamar, mengubah status pelamar dari tertunda menjadi diterima atau ditolak, serta mengirimkan pesan kepada pelamar terkait perubahan status. Halaman detail lamaran admin menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 10 Halaman Detail Lamaran Admin:



Gambar 10. Halaman Detail Lamaran Admin

9. Halaman Pengaturan

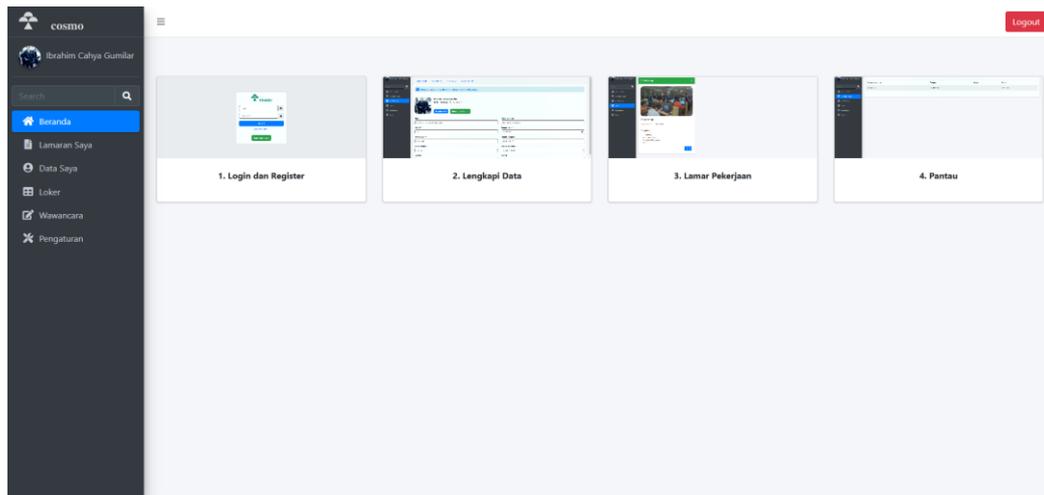
Pada halaman pengaturan, pengguna dapat mengubah informasi akun seperti nama, foto dan kata sandi. Halaman pengaturan menggunakan *service user* dengan *framework Express.js*. Berikut adalah gambar 11 Halaman Pengaturan.



Gambar 11. Halaman Pengaturan

10. Halaman Beranda Pelamar

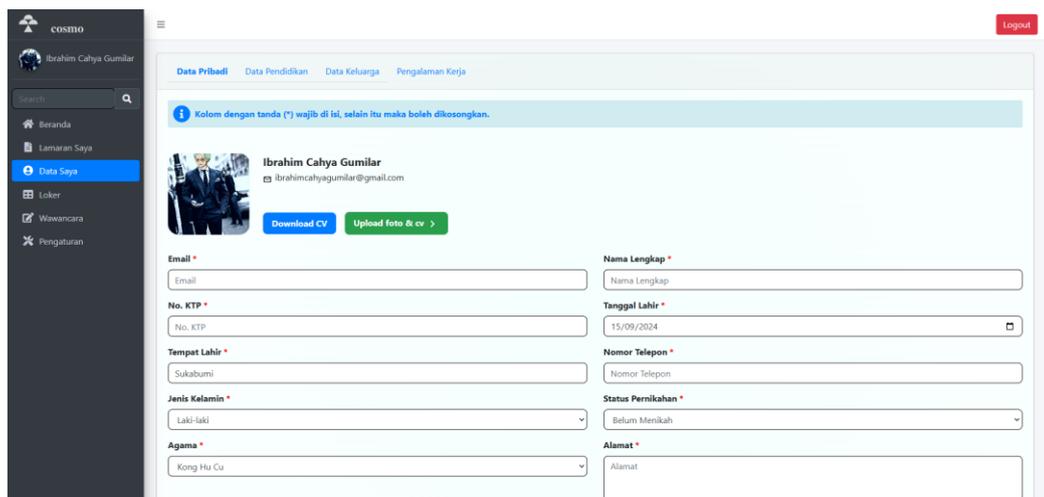
Pada halaman beranda, pelamar dapat melihat langkah-langkah mendaftar pekerjaan melalui sistem. Berikut adalah gambar 12 Halaman Beranda Pelamar:



Gambar 12. Halaman Beranda Pelamar

11. Halaman Data Pribadi Pelamar

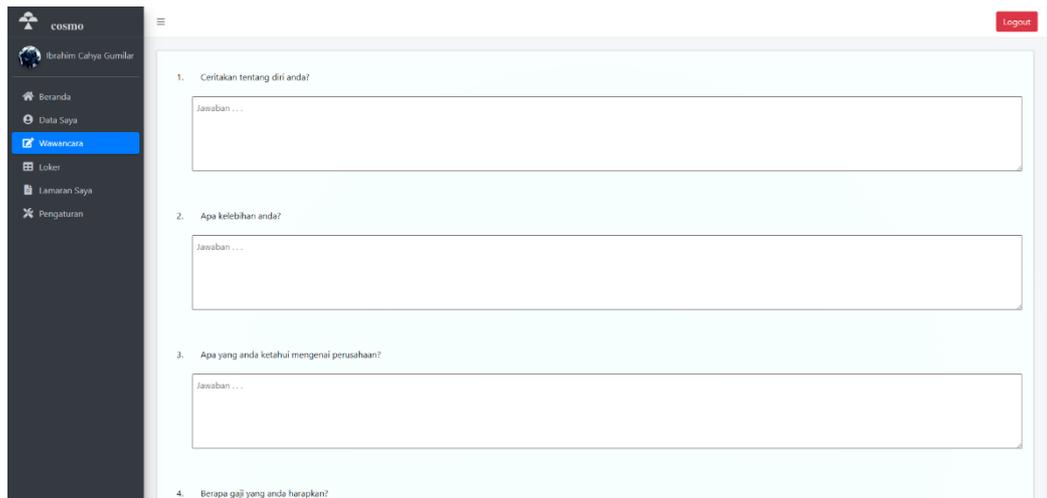
Pada halaman data pribadi, pelamar dapat mengisi formulir yang tersedia sebagai syarat untuk melamar pekerjaan. Formulir ini terdiri dari data pribadi, data pendidikan, data keluarga dan pengalaman kerja. Halaman data pribadi pelamar menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 13 Halaman Data Pribadi Pelamar:



Gambar 13. Halaman Data Pribadi Pelamar

12. Halaman Wawancara Pelamar

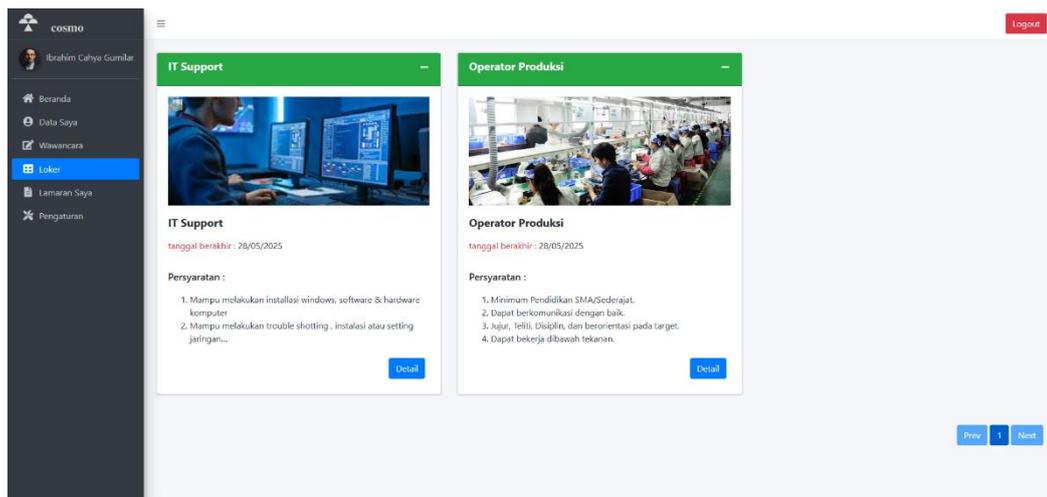
Pada halaman wawancara, pelamar dapat mengisi dan mengubah jawaban dari pertanyaan yang telah disediakan oleh admin. Halaman wawancara pelamar menggunakan *service* wawancara dengan *framework Laravel*. Berikut adalah gambar 14 Halaman Wawancara Pelamar:



Gambar 14. Halaman Wawancara Pelamar

13. Halaman Loker Pelamar

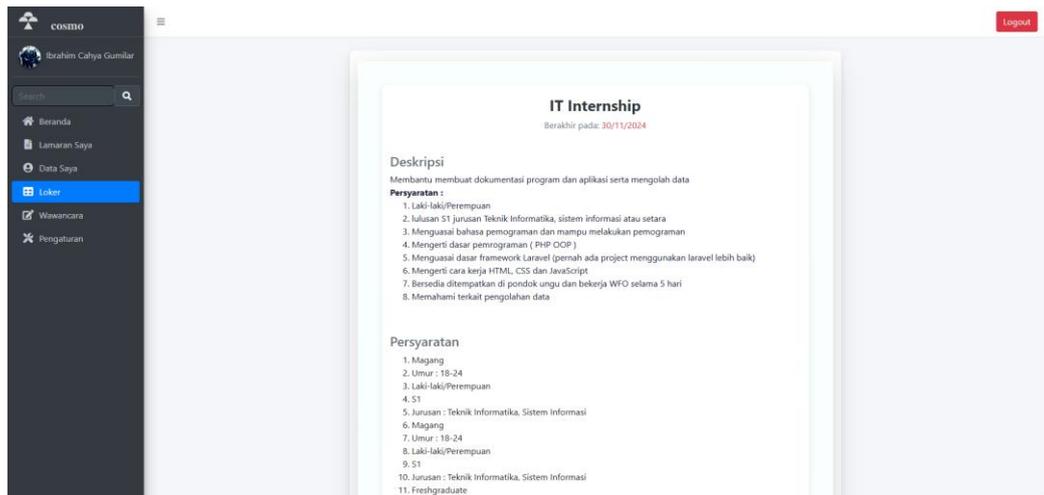
Pada halaman loker, pelamar dapat melihat lowongan pekerjaan yang tersedia. Ketika tombol detail diklik, pelamar akan diarahkan ke halaman detail loker. Halaman loker pelamar menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 15 Halaman Loker Pelamar.



Gambar 15. Halaman Loker Pelamar

14. Halaman Detail Loker Pelamar

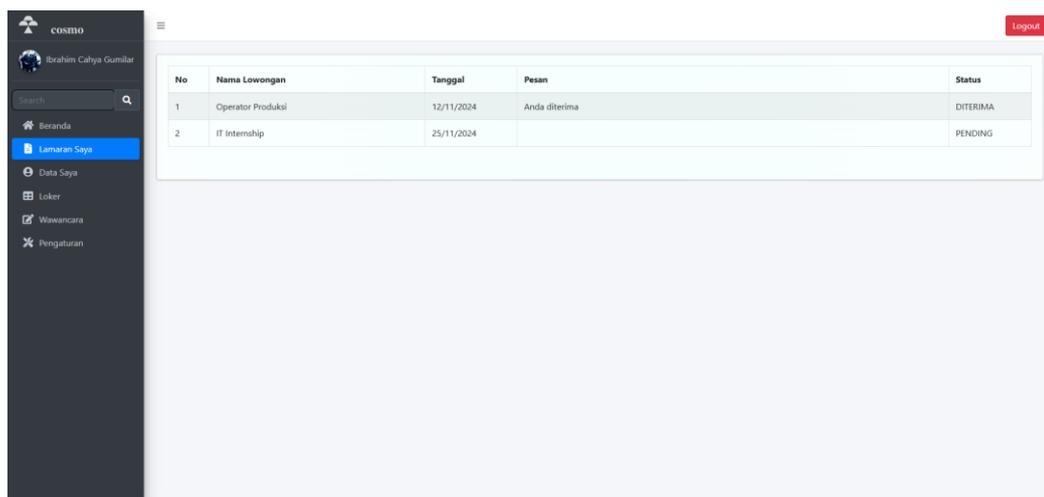
Pada halaman detail loker, pelamar dapat melihat informasi lowongan dan melamar pekerjaan. Halaman detail loker pelamar menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 16 Halaman Detail Loker Pelamar:



Gambar 16. Halaman Detail Loker Pelamar

15. Halaman Lamaran Pelamar

Pada halaman lamaran, pelamar dapat melihat informasi pekerjaan yang telah didaftarkan, seperti nama lowongan, tanggal pendaftaran, pesan, dan status lamaran. Halaman lamaran pelamar menggunakan *service* rekrutmen dengan *framework Express.js*. Berikut adalah gambar 17 Halaman Lamaran Pelamar:



Gambar 17. Halaman Lamaran Pelamar

Pengujian *Black Box*

Pada tahap pengujian, dilakukan dengan memanfaatkan metode *Black Box Testing* yang bertujuan untuk mengidentifikasi kesalahan pada fitur-fitur yang tersedia (Putri et al., 2024). Berikut adalah hasil pengujiannya:

Tabel 1. Pengujian *Black Box* Admin

No	Skenario pengujian	Kasus pengujian	Hasil yang diharapkan	Hasil
1	Login	Masukan email dan password	Berhasil masuk menu beranda	Sesuai
2	Menu loker tambah, ubah, hapus	Mengisi seluruh data untuk menambahkan data, mengubah data, dan menghapus data	Berhasil menambahkan, mengubah, dan menghapus data	Sesuai

3	Menu wawancara tambah, ubah, hapus	Mengisi seluruh data untuk menambahkan data, mengubah data, dan menghapus data	Berhasil menambahkan, mengubah, dan menghapus data	Sesuai
4	Menu lamaran	Menggunakan fitur filter, search, button detail, dan button hapus	Seluruh kasus pengujian berfungsi dengan baik	Sesuai
5	Menu detail lamaran	Klik button terima atau tolak	Berhasil merubah status lamaran	Sesuai
6	Log out	Klik button log out	Keluar dari sistem	Sesuai

Tabel 2. Pengujian Black Box Pelamar

No	Skenario pengujian	Kasus pengujian	Hasil yang diharapkan	Hasil
1	Register	Mengisi seluruh data	Berhasil mendaftar	Sesuai
2	Menu data saya	Mengisi seluruh data	Berhasil memperbarui data	Sesuai
3	Menu wawancara	Mengisi seluruh data	Berhasil memperbarui data	Sesuai
4	Menu loker	Klik button detail	Berhasil membuka menu detail loker	Sesuai
5	Menu detail loker	Klik button daftar	Berhasil daftar	Sesuai

Skala Likert

Dalam penelitian ini, dilakukan survey terhadap 10 responden untuk mengukur pendapat menggunakan skala *Likert* dengan rentang 1 (sangat tidak setuju) hingga 5 (sangat setuju) (Maulana & Dani, 2023). Berikut adalah Tabel 3 skala *Likert*:

Tabel 3. Rata-rata Skala Likert

No	Pertanyaan	Rata-Rata	Interpretasi
1	Kemudahan dalam menggunakan <i>website</i>	4,3	Baik
2	Kecepatan respons website saat diakses	4,5	Baik
3	Efektivitas fitur-fitur yang tersedia pada <i>website</i>	3,9	Netral
4	Kesesuaian fitur <i>website</i> dengan kebutuhan pengguna	4,3	Baik
5	Kepuasan terhadap pengalaman melamar pekerjaan pada <i>website</i>	4,3	Baik

KESIMPULAN

Penelitian ini berhasil mengembangkan sistem informasi *Human Resource Development* berbasis web dengan arsitektur *microservice* untuk meningkatkan efisiensi dan efektivitas proses rekrutmen di PT Cosmo Technology Indonesia. Sebelumnya, proses rekrutmen dilakukan secara manual dengan pengumpulan dokumen fisik yang menyebabkan penumpukan berkas, waktu pemrosesan yang lama, serta biaya tambahan bagi pelamar. Dengan adanya sistem berbasis web, proses rekrutmen menjadi lebih terstruktur dan cepat, memungkinkan pelamar mengajukan lamaran secara online tanpa perlu menyerahkan dokumen fisik. Hasil pengujian *black box* menunjukkan bahwa seluruh fungsi sistem berjalan sesuai kebutuhan, sementara survei dengan skala *Likert* mendapatkan respon positif terhadap peningkatan efisiensi dan kemudahan pengelolaan. Arsitektur *microservice* memberikan manfaat signifikan dalam hal skalabilitas, fleksibilitas, dan reliabilitas, sehingga aplikasi dapat dikembangkan lebih lanjut sesuai dengan kebutuhan perusahaan. Namun, penelitian ini masih memiliki keterbatasan dalam jumlah pengguna yang terlibat dalam pengujian, sehingga penelitian

selanjutnya disarankan untuk dilakukan dalam skala lebih besar serta mengintegrasikan sistem ini dengan platform lain yang digunakan perusahaan agar fungsionalitasnya semakin optimal.

REFERENSI

- Atmojo, S., Utami, R., Dewi, S., & Widhiyanta, N. (2022). Implementasi Sistem-informasi Desa Berbasis Arsitektur Microservices. *Smatika Jurnal*, 12(01), 55–66. <https://doi.org/10.32664/smatika.v12i01.658>
- Cosmo. (n.d.). *LED Lighting Business Unit*. Cosmo.
- Maulana, Y. A., & Dani, H. (2023). Evaluasi Sistem Proteksi Kebakaran Pada Proyek Pembangunan Gedung Rumah Sakit X Di Kota Gresik. *Jurnal Vokasi Teknik Sipil*, 1(2), 106–111. <https://ejournal.unesa.ac.id/index.php/viteks/article/view/55518>
- Panja, E., & Manongga, D. (2023). Perancangan Sistem Informasi Keuangan Berbasis Web Pada Gks Mauliru Menggunakan Metode Rapid Application Development. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(1), 579–584. <https://doi.org/10.36040/jati.v7i1.6401>
- Purwati, N., Fadhlurrahman, O. R., Iswahyuni, D., Kiswati, S., & Faqih, H. (2023). Sistem Informasi Cuti Karyawan Menggunakan Berbasis Web dengan Metode Rapid Application Development (RAD). *Infomatek*, 25, no.1, 61–68. <https://doi.org/10.23969/infomatek.v25i1.7822>
- Putra Permana, D., Ashaury, H., & Nurul Sabrina, P. (2023). Reengineering Arsitektur Monolithic Ke Microservices Pada Website Management Content Mqtv. *Jurnal Locus Penelitian Dan Pengabdian*, 2(9), 924–931. <https://doi.org/10.58344/locus.v2i9.1723>
- Putri, M. R., Ghofur, A., & Azise, N. (2024). Sistem Informasi Pendaftaran Santri Di Pondok Pesantren Al-Wathoniyah 43 Jakarta Utara Berbasis Web Dengan Fitur Whatsapp Gateway. *Jurnal Computer Science and Information Technology (CoSciTech)*, 5(2), 272–281.
- Risnanto, H. (2022). Rancang Bangun Sistem Informasi Layanan Mandiri Perpustakaan Berbasis Arsitektur Microservice. *Repository.Uinjkt.Ac.Id*.