

**Laporan Proyek Pengembangan Aplikasi Web
Sistem Informasi Lowongan Kerja Berbasis Web
Menggunakan Laravel**



DI SUSUN OLEH :

- 1. Dedi Faqih W (225410005)**
- 2. Tirza Sarwono (235410071)**

**PROGRAM STUDI INFORMATIKA KELAS 1
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
(UTDI)
2024**

Laporan Proyek Pengembangan Aplikasi Web Sistem Informasi Lowongan Kerja Berbasis Web Menggunakan Laravel

Dedi Faqih Wahana¹⁾, Tirza Sarwono²⁾

¹⁾ dedi.faqih@students.utdi.ac.id , ²⁾ tirza.sarwono23@students.utdi.ac.id

1. PENDAHULUAN

1.1 Latar Belakang Masalah

Proses rekrutmen merupakan langkah krusial dalam setiap organisasi, di mana perekrutan kandidat yang tepat akan menentukan kualitas dan kesuksesan tim. Namun, banyak perusahaan masih menghadapi tantangan dalam mengelola proses rekrutmen secara efisien, baik dari sisi pelamar maupun tim HR. Proses manual atau sistem yang tidak terintegrasi seringkali menyebabkan kesalahan dalam pemrosesan data, keterlambatan, dan pengalaman pengguna yang buruk. Dengan berkembangnya teknologi, sistem berbasis aplikasi yang dapat membantu memfasilitasi proses rekrutmen menjadi solusi yang sangat dibutuhkan.

Aplikasi rekrutmen berbasis web menjadi solusi untuk mempercepat dan mempermudah interaksi antara pelamar dan tim HR, memungkinkan pelamar untuk melamar pekerjaan dengan lebih mudah dan tim HR untuk mengelola data pelamar secara lebih efisien. Dengan sistem ini, proses seleksi dan pengelolaan data dapat dilakukan secara lebih sistematis, aman, dan cepat.

1.1.1 Rumusan Masalah

- Berdasarkan latar belakang di atas, beberapa permasalahan yang dihadapi dalam proses rekrutmen antara lain:
- 1. Bagaimana cara memfasilitasi pelamar untuk melamar pekerjaan secara efisien dan mengakses informasi pekerjaan yang relevan?
- 2. Bagaimana cara mengelola data pelamar dan pekerjaan secara lebih sistematis dan efisien untuk mempermudah tim HR dalam membuat keputusan?
- 3. Bagaimana memastikan keamanan data pribadi pelamar dan mengelola hak akses pengguna agar hanya pihak yang berwenang yang dapat mengakses atau memodifikasi data sensitif?
- 4. Bagaimana memastikan aplikasi memiliki kinerja yang responsif dan dapat menangani banyak pelamar tanpa mengurangi pengalaman pengguna?

1.2 Tujuan Penelitian

Tujuan dari pengembangan aplikasi rekrutmen ini adalah untuk menciptakan sistem yang dapat mempermudah dan mempercepat proses rekrutmen, baik bagi pelamar maupun tim HR. Pertama, aplikasi ini bertujuan untuk membantu pelamar dalam melihat dan melamar pekerjaan dengan lebih mudah dan cepat, memberikan mereka akses yang jelas dan cepat terhadap berbagai informasi pekerjaan yang relevan, termasuk deskripsi pekerjaan, kualifikasi, dan status lamaran. Selain itu, aplikasi ini diharapkan dapat mempermudah tim HR dalam mengelola data pelamar dan informasi pekerjaan secara lebih efisien dengan sistem yang terintegrasi, sehingga tim HR dapat dengan mudah mengakses, mengelompokkan, dan mengolah data pelamar untuk membuat keputusan perekrutan yang lebih baik dan cepat. Dalam hal keamanan, aplikasi ini dirancang untuk menjamin perlindungan data pribadi pelamar dengan memberikan kontrol akses yang tepat, memastikan hanya administrator yang berwenang yang dapat mengubah atau mengakses data sensitif. Tujuan lainnya adalah untuk meningkatkan efisiensi dan efektivitas

proses rekrutmen dengan mengurangi kesalahan manual yang sering terjadi pada proses berbasis kertas dan mempercepat pemrosesan data, sehingga proses seleksi dapat dilakukan dengan lebih cepat dan tepat. Terakhir, aplikasi ini akan menyediakan antarmuka yang user-friendly dan responsif, memberikan pengalaman pengguna yang optimal baik bagi pelamar yang ingin melamar pekerjaan maupun tim HR yang bertugas mengelola proses rekrutmen, menjadikan sistem ini mudah digunakan dan diakses oleh berbagai jenis pengguna.

1.3 Batasan Masalah

Untuk menjaga fokus pada pengembangan aplikasi rekrutmen yang sesuai dengan kebutuhan utama, batasan masalah dalam proyek ini adalah sebagai berikut:

Fitur yang Dikembangkan:

1. Bagi Pelamar:
 - Fitur untuk melihat daftar pekerjaan berdasarkan kategori, deskripsi pekerjaan, dan status.
 - Formulir pendaftaran pekerjaan yang memungkinkan pelamar untuk mengajukan lamaran secara online.
 - Tampilan status lamaran untuk melihat apakah pelamar diterima, dijadwalkan untuk wawancara, atau ditolak.
2. Bagi Admin (HR):
 - Pengelolaan data pelamar dan pekerjaan melalui dashboard.
 - Fitur pencarian dan filter untuk mempermudah pencarian pelamar dan pekerjaan.
 - Backup dan restore data pekerjaan dan pelamar.
 - Pengaturan hak akses untuk administrator dan pelamar.

Fitur yang Tidak Dikembangkan:

1. Sistem Penilaian Otomatis:
 - Aplikasi ini tidak akan mengotomatiskan penilaian kualifikasi pelamar.
2. Komunikasi Internal:
 - Tidak ada fitur percakapan atau chat antar pelamar dan admin.
3. Integrasi dengan Media Sosial atau Platform Pihak Ketiga:
 - Fitur ini tidak akan mencakup integrasi dengan platform seperti LinkedIn atau portal karir eksternal.
4. Fitur Pengelolaan Gaji atau Pekerjaan Pasca-Perekrutan:
 - Sistem ini hanya akan fokus pada proses rekrutmen, tanpa mencakup pengelolaan gaji atau administrasi pasca-rekrutmen.

■
■

2. PERANCANGAN SISTEM

Pada bagian perancangan, dijelaskan tentang gambaran umum pembangunan aplikasi rekrutmen, analisis sistem yang sedang berjalan, serta sistem yang diusulkan untuk mendukung proses bisnis rekrutmen secara lebih efisien. Perancangan ini mencakup desain database, alur data, dan struktur antarmuka pengguna yang diharapkan dapat meningkatkan pengalaman pengguna baik bagi pelamar maupun tim HR.

■

2.1 Rancangan Semula/Awal

Aplikasi rekrutmen ini dirancang untuk memfasilitasi proses perekrutan secara efisien dengan memberikan kemudahan kepada pelamar untuk melamar pekerjaan serta memudahkan tim HR dalam mengelola data pelamar dan pekerjaan. Pengguna (pelamar dan admin) dapat berinteraksi dengan sistem melalui antarmuka yang sederhana dan responsif.

Secara keseluruhan, aplikasi ini akan memiliki dua peran utama, yaitu:

1. Pelamar: Pengguna yang dapat mengakses daftar pekerjaan, mengisi formulir lamaran, dan melacak status lamaran.
2. Administrator (HR): Pengguna yang bertanggung jawab mengelola data pelamar, pekerjaan, dan wawancara.

2.1.1 Rancangan Database

Berikut adalah desain database untuk aplikasi rekrutmen yang mencakup tiga tabel utama: Tabel Admin, Tabel Pelamar, dan Tabel Pekerjaan. Desain ini mencakup relasi antar tabel dan SQL syntax untuk pembuatan tabel tersebut.

1. Tabel Admin

Tabel ini berfungsi untuk menyimpan data administrator yang mengelola aplikasi rekrutmen.

Format Tabel :

Kolom	Tipe Data	Keterangan
id	INT	Primary Key, Auto Increment
name	VARCHAR(255)	Nama administrator
email	VARCHAR(255)	Email administrator, harus unik
password	VARCHAR(255)	Password administrator
created_at	TIMESTAMP	Waktu pembuatan data admin
updated_at	TIMESTAMP	Waktu pembaruan data admin

SQL Syntax:

```
CREATE TABLE admins (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(255) NOT NULL,
  email VARCHAR(255) UNIQUE NOT NULL,
  password VARCHAR(255) NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
  CURRENT_TIMESTAMP
);
```

2. Tabel Pekerjaan

Tabel ini menyimpan informasi mengenai pekerjaan yang tersedia untuk pelamar.

Format Tabel :

Kolom	Tipe Data	Keterangan
id_pekerjaan	INT	Primary Key, Auto Increment
nama	VARCHAR(255)	Nama posisi pekerjaan
status	ENUM('aktif', 'non-aktif')	Status pekerjaan (aktif atau non-aktif)
created_at	TIMESTAMP	Waktu pembuatan data pekerjaan
updated_at	TIMESTAMP	Waktu pembaruan data pekerjaan

SQL Syntax :

```
CREATE TABLE pekerjaan (
  id_pekerjaan INT AUTO_INCREMENT PRIMARY KEY,
  nama VARCHAR(255) NOT NULL,
```

```

status ENUM('aktif', 'non-aktif') DEFAULT 'aktif',
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP
);

```

3. Tabel Pelamar

Tabel ini berfungsi untuk menyimpan data pelamar yang melamar pekerjaan yang tersedia.

Format Tabel :

Kolom	Tipe Data	Keterangan
id_pelamar	INT	Primary Key, Auto Increment
nama_lengkap	VARCHAR(255)	Nama lengkap pelamar
tgl_lahir	DATE	Tanggal lahir pelamar
alamat	TEXT	Alamat pelamar
kode_pos	VARCHAR(10)	Kode pos pelamar
email	VARCHAR(255)	Email pelamar, harus unik
jenis_kelamin	ENUM('L', 'P')	Jenis kelamin pelamar (L = Laki-laki, P = Perempuan)
no_tlp	VARCHAR(20)	Nomor telepon pelamar
id_pekerjaan	INT	Foreign Key mengacu pada pekerjaan
lulusan	VARCHAR(255)	Lulusan pelamar
berkas	VARCHAR(255)	Nama file berkas pelamar (misal PDF)
status	ENUM('pending', 'diterima', 'ditolak')	Status lamaran pelamar
created_at	TIMESTAMP	Waktu pembuatan data pelamar
updated_at	TIMESTAMP	Waktu pembaruan data pelamar

SQL Syntax :

```

CREATE TABLE pelamar (
  id_pelamar INT AUTO_INCREMENT PRIMARY KEY,
  nama_lengkap VARCHAR(255) NOT NULL,
  tgl_lahir DATE NOT NULL,
  alamat TEXT,
  kode_pos VARCHAR(10),
  email VARCHAR(255) UNIQUE NOT NULL,
  jenis_kelamin ENUM('L', 'P'),
  no_tlp VARCHAR(20),
  id_pekerjaan INT,
  lulusan VARCHAR(255),
  berkas VARCHAR(255), -- nama file pdf
  status ENUM('pending', 'diterima', 'ditolak') DEFAULT 'pending',
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

```

```

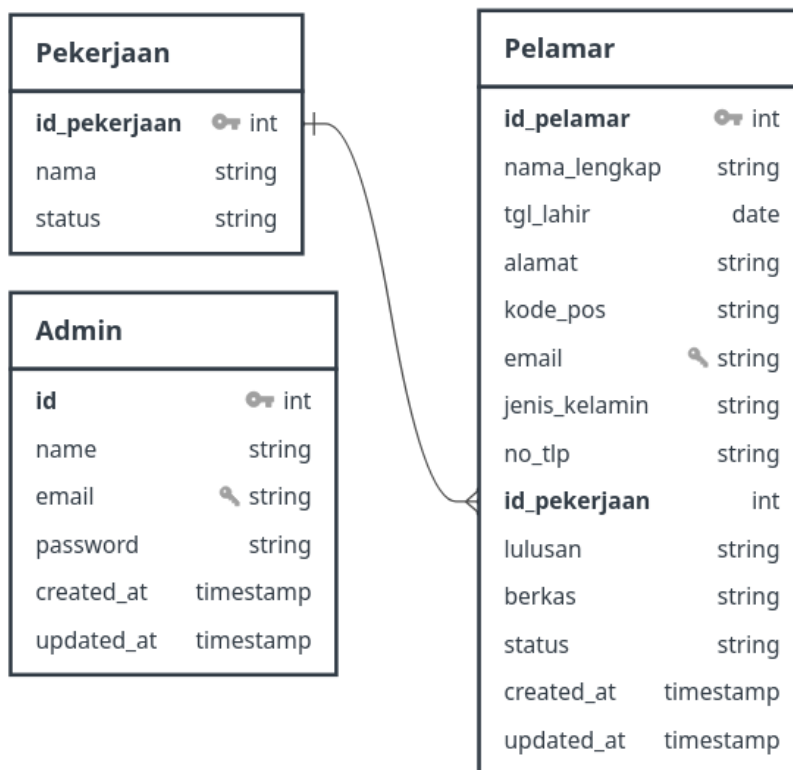
        updated_at    TIMESTAMP    DEFAULT    CURRENT_TIMESTAMP    ON    UPDATE
CURRENT_TIMESTAMP,
        FOREIGN KEY (id_pekerjaan) REFERENCES pekerjaan(id_pekerjaan) ON DELETE
SET NULL
);

```

Relasi Antar Tabel :

- Pelamar memiliki relasi dengan Pekerjaan melalui id_pekerjaan. Setiap pelamar dapat melamar untuk satu pekerjaan, yang menunjukkan relasi many-to-one antara pelamar dan pekerjaan.
- Pekerjaan memiliki status "aktif" atau "non-aktif", sehingga hanya pekerjaan yang "aktif" yang akan tersedia bagi pelamar.
- Admin bertugas mengelola pekerjaan dan pelamar, tetapi tidak ada relasi langsung antara admin dan pelamar atau pekerjaan dalam hal struktur tabel ini.

Diagram Relasi Tabel :

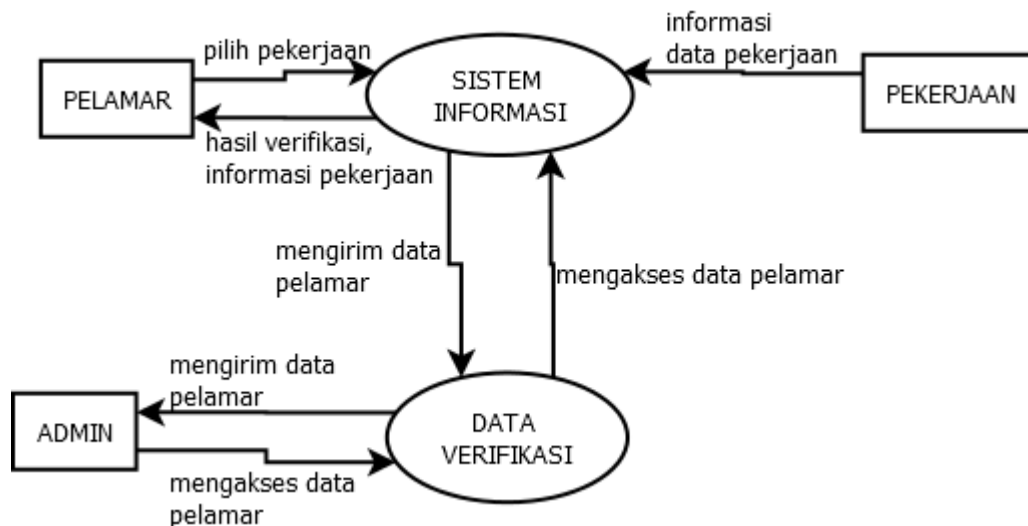


Keterangan : Dengan rancangan database ini, aplikasi rekrutmen dapat mengelola data pelamar, pekerjaan, dan administrator dengan efisien. Relasi antar tabel juga memastikan bahwa data pelamar dapat terhubung dengan pekerjaan yang dilamar, sementara admin memiliki kontrol penuh terhadap pekerjaan dan pelamar. Struktur tabel dan query SQL yang disediakan dapat langsung digunakan untuk pembuatan database.

2.1.2 Data flow diagram (DFD) level 0

Data Flow Diagram (DFD) Level 0 menggambarkan gambaran umum dari sistem, menunjukkan bagaimana data mengalir di antara entitas eksternal (seperti pelamar dan admin) dan sistem rekrutmen. DFD Level 0 ini akan memberikan pemahaman mengenai aliran data dan proses utama yang terlibat dalam aplikasi rekrutmen.

DFD level 0 :



Keterangan : Diagram DFD ini menggambarkan alur data pada sistem rekrutmen online terkait proses pemilihan pekerjaan dan verifikasi data. Pelamar berinteraksi dengan Sistem Informasi untuk memilih pekerjaan, di mana sistem akan mengambil informasi data pekerjaan dari entitas Pekerjaan dan menampilkannya kepada pelamar. Selanjutnya, Sistem Informasi mengirimkan data pelamar ke Data Verifikasi untuk proses validasi. Admin dapat mengakses data pelamar dari Data Verifikasi dan mengirimkan kembali data tersebut setelah verifikasi selesai. Alur ini menunjukkan hubungan antara pelamar, admin, sistem informasi, dan data pekerjaan, di mana setiap komponen berperan dalam proses pengelolaan data lamaran kerja.

○

2.2 Realisasi

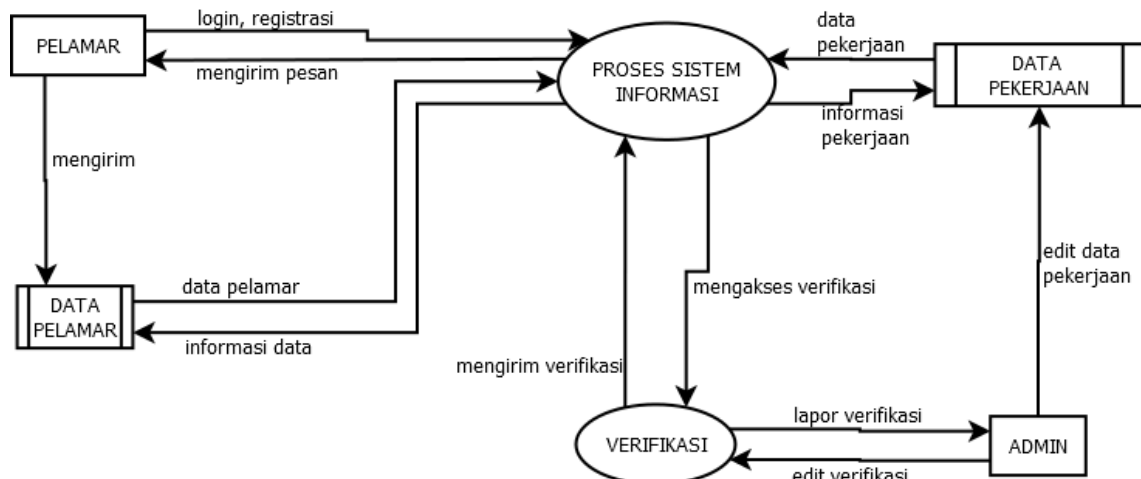
Berikut adalah DFD Realisasi yang lebih terperinci dan menggambarkan proses-proses dalam sistem rekrutmen berdasarkan tabel yang telah diimplementasikan:

Proses Pelamar :

- **Pilih Pekerjaan:** Pelamar memilih pekerjaan yang sesuai dan tersedia dalam sistem, yang diambil dari Tabel Pekerjaan.
- **Isi Formulir Lamaran:** Pelamar mengisi formulir lamaran yang mencakup data pribadi dan berkas yang akan disimpan dalam Tabel Pelamar.
- **Kirim Lamaran:** Setelah mengisi formulir, pelamar mengirimkan lamaran yang akan disimpan dalam Tabel Pelamar.

Proses Admin :

- **Manajemen Pelamar:** Admin dapat melihat data pelamar dan memperbarui status mereka (misalnya, diterima atau ditolak).
- **Tabel Pekerjaan & Tabel Admin:** Admin dapat menambah, mengedit, atau menghapus pekerjaan yang tersedia di Tabel Pekerjaan



Keterangan : Dalam diagram ini, terdapat beberapa komponen utama, yaitu Pelamar, yang bertugas untuk melakukan login, registrasi, serta mengirimkan data pribadi dan lamaran; Proses Sistem Informasi, yang merupakan inti dari sistem yang memproses data yang masuk dan keluar; Data Pekerjaan, yang menyimpan informasi terkait lowongan pekerjaan yang tersedia; Data Pelamar, yang menyimpan data pribadi pelamar; Verifikasi, yang berfungsi untuk memeriksa kelengkapan dan keabsahan data lamaran yang diterima; serta Admin, yang bertanggung jawab untuk mengelola data pekerjaan dan memantau hasil verifikasi. Alur data dimulai dengan pelamar yang melakukan login atau registrasi, kemudian mengirimkan data pribadinya ke Proses Sistem Informasi. Data pelamar akan disimpan dalam Data Pelamar, dan data lowongan pekerjaan diambil dari Data Pekerjaan untuk ditampilkan kepada pelamar. Pelamar memilih lowongan pekerjaan yang diinginkan dan mengirimkan lamaran yang kemudian diproses oleh Verifikasi. Hasil verifikasi akan diteruskan ke Admin yang dapat melakukan tindakan lebih lanjut, seperti memberikan tanggapan terhadap pelamar atau mengelola data pekerjaan.

2.2.1 as

3. TEKNOLOGI

3.1 Teknologi/Framework yang Digunakan

Dalam pembuatan Sistem Informasi Lowongan Kerja Berbasis Web, teknologi dan framework yang digunakan antara lain:

1. Laravel
 - Laravel dipilih sebagai framework PHP karena kemampuannya dalam mengelola kode yang terstruktur, aman, dan efisien.
 - Laravel memiliki fitur migration, routing, dan Eloquent ORM yang memudahkan pengembangan aplikasi berbasis database.
 - Kemudahan dalam autentikasi pengguna dan middleware menjadikan Laravel sangat cocok untuk sistem seperti ini.
2. Bootstrap
 - Digunakan untuk mengatur tampilan antarmuka pengguna (UI) agar responsif dan menarik.
 - Bootstrap mempermudah pengembangan frontend tanpa harus membuat CSS dari awal.
3. MySQL
 - MySQL digunakan sebagai database untuk menyimpan data pelamar, admin, dan pekerjaan.

- Relasi antar tabel dapat dikelola dengan baik menggunakan MySQL dan Laravel ORM.
4. XAMPP
- Digunakan sebagai server lokal untuk menjalankan Laravel dan MySQL secara offline.

4. IMPLEMENTASI

4.1 Implementasi

1. Kode Migrasi Database

Kode berikut menunjukkan implementasi tabel database menggunakan Laravel Migration :

a. Tabel Admin

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
use Illuminate\Support\Facades\DB;
use Illuminate\Support\Facades\Hash;

class CreateAdminsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('admins', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->string('email')->unique();
            $table->string('password');
            $table->timestamps();
        });
        // Insert data admin
        DB::table('admins')->insert([
            'name' => 'Admin Pertama',
            'email' => 'admin@example.com',
            'password' => Hash::make('password123'),
            'created_at' => now(),
            'updated_at' => now(),
        ]);
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('admins');
    }
}
```

```

    }
}

```

b. Tabel Pekerjaan

```

<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
use Illuminate\Support\Facades\DB;

class CreatePekerjaanTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('pekerjaan', function (Blueprint $table) {
            $table->id('id_pekerjaan');
            $table->string('nama');
            $table->enum('status', ['aktif', 'non-aktif'])-
>default('aktif');
            $table->timestamps();
        });
        // Insert data pekerjaan
        DB::table('pekerjaan')->insert([
            ['nama' => 'Software Engineer', 'status' => 'aktif',
'created_at' => now(), 'updated_at' => now()],
            ['nama' => 'Data Analyst', 'status' => 'aktif', 'created_at'
=> now(), 'updated_at' => now()],
            ['nama' => 'Product Manager', 'status' => 'aktif',
'created_at' => now(), 'updated_at' => now()],
        ]);
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('pekerjaan');
    }
}

```

c. Tabel Pelamar

```

<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
use Illuminate\Support\Facades\DB;

```

```

class CreatePelamarTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('pelamar', function (Blueprint $table) {
            $table->id('id_pelamar');
            $table->string('nama_lengkap');
            $table->date('tgl_lahir');
            $table->text('alamat')->nullable();
            $table->string('kode_pos', 10)->nullable();
            $table->string('email')->unique();
            $table->enum('jenis_kelamin', ['L', 'P']);
            $table->string('no_tlp', 20)->nullable();
            $table->unsignedBigInteger('id_pekerjaan')->nullable(); //
foreign key ke id_pekerjaan di tabel pekerjaan
            $table->foreign('id_pekerjaan')->references('id_pekerjaan')->
on('pekerjaan')->onDelete('set null');
            $table->string('lulusan');
            $table->string('berkas'); // file berkas PDF
            $table->enum('status', ['pending', 'diterima', 'ditolak'])->
default('pending');
            $table->timestamps();
        });
        // Insert data pelamar
        DB::table('pelamar')->insert([
            [
                'nama_lengkap' => 'John Doe',
                'tgl_lahir' => '1990-01-01',
                'alamat' => 'Jl. Example No. 1',
                'kode_pos' => '12345',
                'email' => 'johndoe@example.com',
                'jenis_kelamin' => 'L',
                'no_tlp' => '081234567890',
                'id_pekerjaan' => 1, // Software Engineer
                'lulusan' => 'S1 Teknik Informatika',
                'berkas' => 'johndoe_cv.pdf',
                'created_at' => now(),
                'updated_at' => now(),
            ],
            [
                'nama_lengkap' => 'Jane Smith',
                'tgl_lahir' => '1992-02-02',
                'alamat' => 'Jl. Example No. 2',
                'kode_pos' => '54321',
                'email' => 'janesmith@example.com',
                'jenis_kelamin' => 'P',
            ],
        ]);
    }
}

```

```

        'no_tlp' => '089876543210',
        'id_pekerjaan' => 2, // Data Analyst
        'lulusan' => 'S1 Statistik',
        'berkas' => 'janesmith_cv.pdf',
        'created_at' => now(),
        'updated_at' => now(),
    ],
    [
        'nama_lengkap' => 'Michael Johnson',
        'tgl_lahir' => '1988-03-03',
        'alamat' => 'Jl. Example No. 3',
        'kode_pos' => '67890',
        'email' => 'michaeljohnson@example.com',
        'jenis_kelamin' => 'L',
        'no_tlp' => '087654321098',
        'id_pekerjaan' => 3, // Product Manager
        'lulusan' => 'S2 Manajemen Bisnis',
        'berkas' => 'michaeljohnson_cv.pdf',
        'created_at' => now(),
        'updated_at' => now(),
    ]
    ]);

}

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('pelamar');
}
}

```

Dengan menggunakan Laravel, Bootstrap, dan MySQL, sistem informasi rekrutmen ini dapat berjalan secara fungsional dengan fitur manajemen pekerjaan, pendaftaran pelamar, dan pengelolaan data rekrutmen. Relasi antar tabel mendukung integrasi data yang efisien, sementara antarmuka pengguna dibuat responsif dan mudah digunakan.

5. MENJALANKAN APLIKASI

5.1 Langkah Langkah Menjalankan Aplikasi

Berikut adalah langkah langkah agar aplikasi dapat dijalankan :

1. Persiapan Lingkungan Pengembangan

- Install XAMPP

Unduh dan install XAMPP untuk menjalankan server Apache dan MySQL secara lokal.

Link unduh: <https://www.apachefriends.org>

- Install Composer

Composer digunakan untuk mengelola dependensi Laravel.

Unduh dan install Composer: <https://getcomposer.org>.

- Install Laravel

Install Laravel menggunakan Composer:

composer global require laravel/installer

2. Persiapkan Database

- Jalankan XAMPP

Jalankan XAMPP dan aktifkan Apache serta MySQL.

- Buat Database

Akses phpMyAdmin melalui <http://localhost/phpmyadmin> dan buat database baru, misalnya db_rekrutmen.

- Konfigurasi .env Laravel

Buka file .env di root folder Laravel dan sesuaikan konfigurasi database:

DB_CONNECTION=mysql

DB_HOST=127.0.0.1

DB_PORT=3306

DB_DATABASE=db_rekrutmen

DB_USERNAME=root

DB_PASSWORD=

3. Clone atau Buat Proyek Laravel

- Clone Proyek

Jika aplikasi sudah ada di repository, gunakan Git untuk meng-clone proyek:

```
git clone <repository-link>
```

```
cd <nama-folder-proyek>
```

- Buat Proyek Baru (Jika diperlukan)

Jika membuat aplikasi baru, gunakan perintah:

```
laravel new sistem-rekrutmen
```

```
cd sistem-rekrutmen
```

4. Migrasi Database

Jalankan perintah berikut untuk membuat tabel di database menggunakan Laravel migration :

```
php artisan migrate
```

5. Menjalankan Aplikasi di Server Lokal

Jalankan server Laravel dengan perintah:

```
php artisan serve --host=103.150.197.76 --port=8080
```

--host=103.150.197.76: Menetapkan IP publik untuk server.

--port=8080: Menetapkan port yang digunakan, dalam hal ini port 8080.

6. Akses Aplikasi melalui IP

Akses Aplikasi di Browser

Buka browser dan akses aplikasi melalui alamat IP yang sudah ditentukan:

<http://103.150.197.76:8080/>

Keterangan :

Aplikasi rekrutmen akan muncul, yang memungkinkan admin untuk mengelola pekerjaan dan pelamar, serta pelamar untuk mendaftar pekerjaan yang tersedia.

7. Pengaturan Firewall dan Keamanan Server

- Buka Port 8080 di Firewall

Jika aplikasi di-hosting di server atau komputer lain, pastikan port 8080 terbuka di firewall sehingga dapat diakses dari luar jaringan lokal.

- Keamanan Server

Pastikan server Anda aman, terutama jika aplikasi ini dipublikasikan secara online. Gunakan HTTPS dan enkripsi data, serta pastikan server dan aplikasi terupdate.

8. Cek Server pada Jaringan Eksternal
Verifikasi Akses dari Jaringan Lain

Pastikan aplikasi dapat diakses dari jaringan luar dengan mencoba mengakses IP <http://103.150.197.76:8080/> dari perangkat lain yang tidak terhubung ke jaringan lokal.

Dengan melakukan langkah-langkah ini, aplikasi Laravel dapat dijalankan dan diakses di server menggunakan IP publik, memungkinkan admin dan pelamar untuk mengelola dan mengakses aplikasi rekrutmen.

6. KESIMPULAN DAN SARAN

6.1 Kesimpulan

Pada proyek pengembangan aplikasi web Sistem Informasi Lowongan Kerja Berbasis Web Menggunakan Laravel, telah berhasil dibangun sebuah sistem yang mempermudah proses rekrutmen secara efisien. Sistem ini memungkinkan admin untuk mengelola data pekerjaan, pelamar, serta memantau status pendaftaran pelamar dengan mudah. Dengan memanfaatkan framework Laravel, aplikasi ini memiliki struktur yang terorganisir dengan baik, pengelolaan database yang efisien, serta antarmuka pengguna yang mudah digunakan.

Fitur utama dari aplikasi ini mencakup:

- Pengelolaan tabel admin untuk otentikasi dan pengelolaan data.
- Pengelolaan tabel pelamar untuk menyimpan informasi pribadi dan status pendaftaran.
- Pengelolaan tabel pekerjaan untuk mencatat posisi yang tersedia dan status aktif/non-aktif.

Selain itu, sistem ini memiliki beberapa fitur non-fungsional, seperti keamanan data pelamar dan admin, ketersediaan sistem yang selalu terjaga, dan antarmuka pengguna yang ramah dan mudah digunakan.

•

6.2 Saran

Untuk meningkatkan kinerja dan keamanan aplikasi, beberapa hal yang dapat di sarankan adalah :

1. Peningkatan Keamanan: Menambahkan autentikasi dua faktor (2FA) untuk meningkatkan keamanan akses.
2. Optimasi Performa Aplikasi: Implementasi caching untuk mempercepat respon aplikasi, terutama saat data pelamar dan pekerjaan meningkat.
3. Peningkatan Fitur Pengelolaan Pelamar: Menambahkan fitur pencarian dan filter untuk memudahkan admin memantau status pelamar.
4. Responsivitas Desain: Meningkatkan desain responsif agar aplikasi optimal di berbagai perangkat.
5. Monitoring dan Pemeliharaan Sistem: Melakukan pemeliharaan dan monitoring rutin untuk memperbaiki bug dan memperbarui fitur.

Dengan perbaikan tersebut, aplikasi dapat memberikan pengalaman yang lebih baik dan mendukung efisiensi proses rekrutmen.