Laporan Proyek: Backend Service untuk Screening Kandidat Berbasis AI

Informasi Kandidat

- **Nama Lengkap:** Rafli Ramadhani
- **Alamat Email:** raflirama7446@gmail.com

Link Repositori

- https://github.com/zendParadox/ backend-ai-screener

1. Pendekatan & Desain Sistem

Rencana Awal

Tantangan ini saya pecah menjadi beberapa komponen utama:

1. **API Layer:** Membuat RESTful API yang jelas untuk interaksi dari luar.
2. **Proses Asynchronous:** Mengidentifikasi bahwa evaluasi AI akan memakan waktu, sehingga perlu dipisahkan dari alur request-response HTTP utama.
3. **RAG (Retrieval-Augmented Generation):** Merancang alur untuk menyimpan dan mengambil dokumen referensi (ground truth) untuk memberikan konteks yang akurat kepada LLM.
4. **Integrasi LLM:** Menghubungkan sistem dengan layanan AI generatif untuk melakukan analisis akhir.

Desain Sistem dan Database

API Endpoints: Sistem ini diekspos melalui tiga endpoint utama:

- `POST /upload`: Menerima file `multipart/form-data` (CV dan Laporan), menyimpannya di server, dan mengembalikan nama file sebagai ID unik.
- `POST /evaluate`: Menerima ID file dan detail pekerjaan, lalu menambahkan tugas evaluasi ke dalam antrian (queue) dan segera mengembalikan `job_id`.
- `GET /result/{id}`: Memungkinkan klien untuk memeriksa status pekerjaan (queued, processing, completed, failed) dan mengambil hasil akhir jika sudah selesai.

Antrian Pekerjaan (Job Queue): Untuk menangani proses evaluasi yang berjalan lama, saya menggunakan **BullMQ** dengan **Redis** sebagai backend. Pendekatan ini memastikan bahwa endpoint `/evaluate` dapat memberikan respons dengan cepat tanpa harus menunggu panggilan API ke LLM selesai. Ini juga membangun pondasi yang kuat untuk skalabilitas dan penanganan kegagalan di masa depan.

Database Vektor: Saya memilih **Qdrant** sebagai vector database. Dokumen-dokumen referensi seperti Deskripsi Pekerjaan, Case Study Brief, dan Rubrik Penilaian di-ingest ke dalam sebuah *collection* di Qdrant. Ini memungkinkan sistem untuk melakukan pencarian semantik yang efisien guna menemukan konteks yang paling relevan untuk setiap evaluasi.

Integrasi LLM

Pilihan LLM: Saya menggunakan model **`gemini-2.5-flash`** dari Google AI. Model ini dipilih karena menawarkan keseimbangan yang baik antara kecepatan, kemampuan mengikuti instruksi kompleks (seperti menghasilkan format JSON yang ketat), dan efisiensi biaya.

Strategi RAG (Retrieval-Augmented Generation): Alur RAG diimplementasikan sebagai berikut:

1. Saat evaluasi dimulai, konten dari CV kandidat diubah menjadi vektor embedding menggunakan model `Xenova/all-MiniLM-L6-v2` yang berjalan secara lokal.
2. Vektor ini digunakan untuk melakukan pencarian di Qdrant untuk mengambil dokumen referensi yang paling relevan.
3. Kumpulan dokumen yang relevan ini (konteks) kemudian disuntikkan ke dalam *prompt* yang dikirim ke Gemini.

Strategi Prompting: Prompt dirancang dengan sangat hati-hati untuk memastikan output yang konsisten dan terstruktur:

1. **Persona:** AI diberi peran sebagai "Asisten HR Ahli".
2. **Instruksi Ketat:** Memberikan instruksi yang jelas untuk menggunakan dokumen konteks sebagai satu-satunya sumber kebenaran.
3. **Struktur Output:** Mendefinisikan skema JSON yang wajib diikuti.
4. **Penyisipan Data:** Menyediakan konteks dari RAG dan data dari CV/Laporan kandidat secara terpisah dengan penanda yang jelas.

2. Hasil & Refleksi

Hasil: Arsitektur asinkron dengan BullMQ bekerja dengan sangat baik. Sistem mampu menerima pekerjaan, memprosesnya di latar belakang, dan menyajikan hasilnya tanpa memblokir klien. Alur RAG juga berhasil mengambil konteks yang relevan, yang secara signifikan meningkatkan kualitas output dari Gemini dibandingkan tanpa konteks.

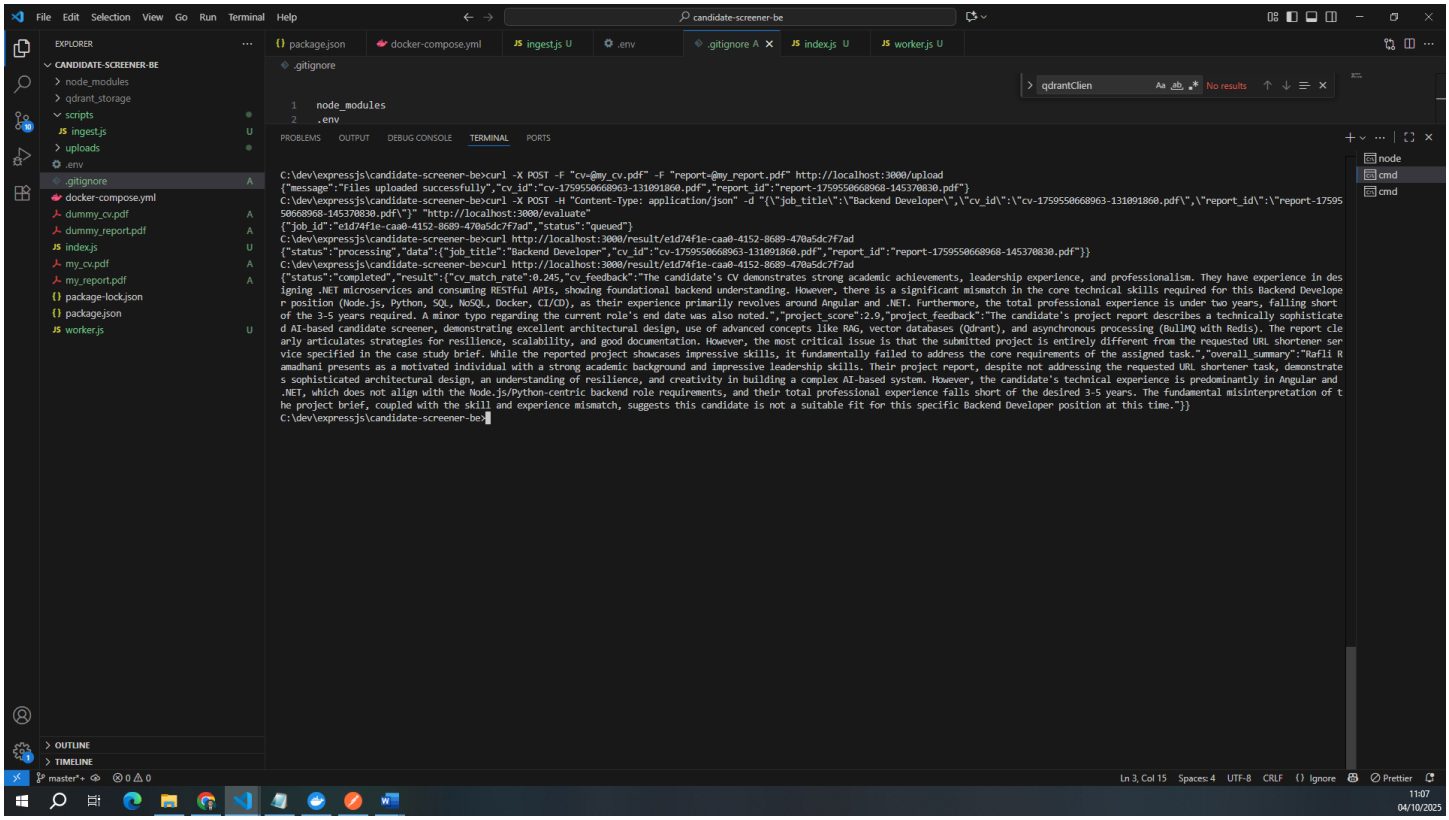
Refleksi: Tantangan terbesar adalah memastikan kompatibilitas versi antara semua komponen (server Qdrant, klien Node.js, dll.). Selain itu, kualitas embedding sangat menentukan keberhasilan RAG. Menggunakan model embedding yang berjalan secara lokal memberikan keuntungan dari segi privasi dan biaya, meskipun memerlukan waktu untuk inisialisasi pada proses pertama.

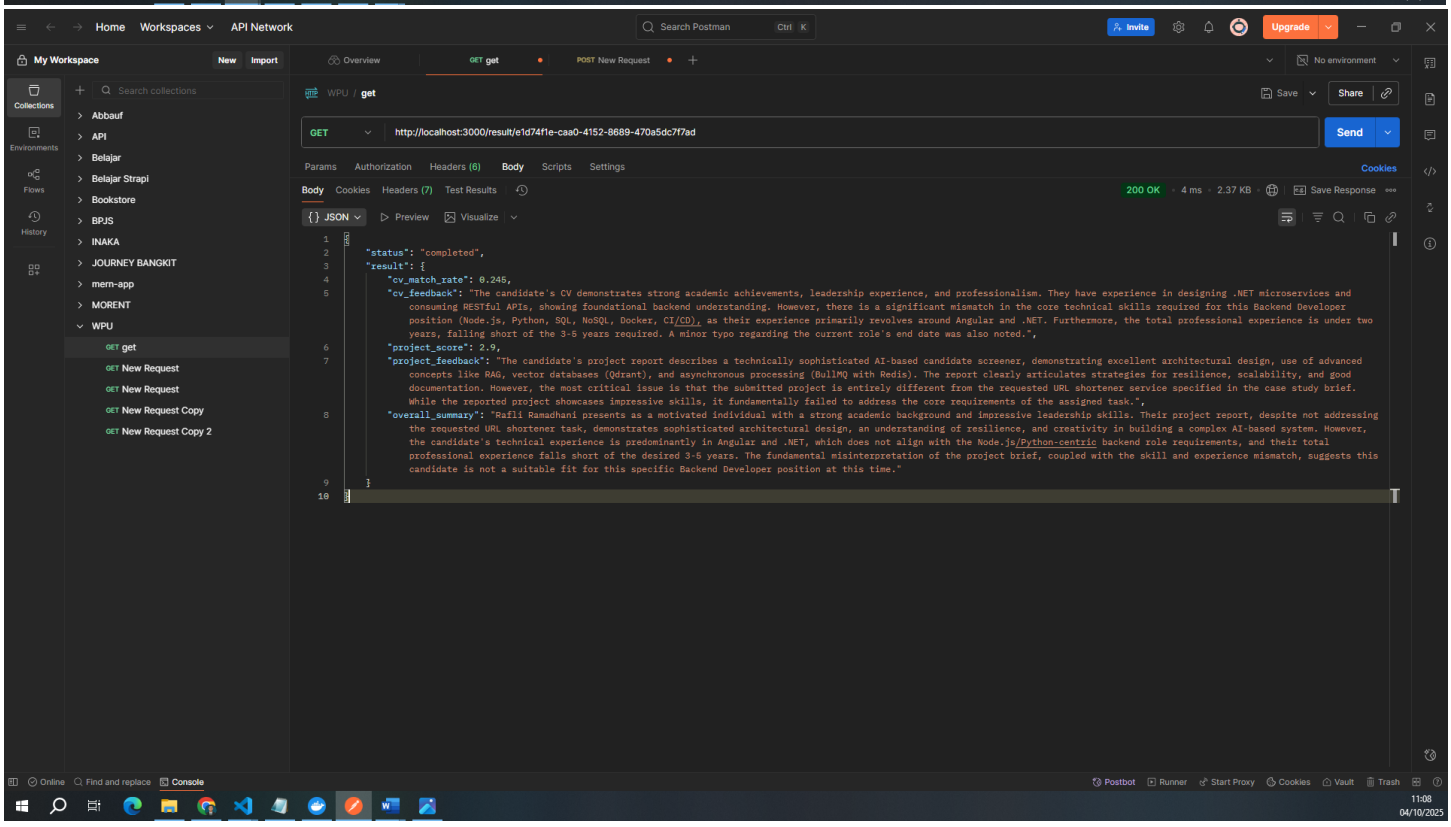
3. Peningkatan di Masa Depan

Dengan lebih banyak waktu, beberapa area yang dapat ditingkatkan adalah:

- **Error Handling:** Mengimplementasikan logika *retry* dengan *exponential backoff* yang lebih canggih untuk panggilan API ke Gemini.
- **Testing:** Menambahkan unit test dan integration test untuk memastikan keandalan setiap komponen.
- **Frontend:** Membangun antarmuka pengguna sederhana untuk memudahkan proses upload dan pengecekan hasil.

4. Screenshots of Real Responses


File Edit Selection View Go Run Terminal Help
candidate-screener-be
EXPLORER
CANDIDATE-SCREENER-BE
node_modules
qdrant_storage
scripts
ingest.js
uploads
.env
.gitignore
docker-compose.yml
dummy_cv.pdf
dummy_report.pdf
index.js
my_cv.pdf
my_report.pdf
package-lock.json
package.json
worker.js
package.json
docker-compose.yml
ingest.js U
.env
.gitignore A
index.js U
worker.js U
.gitignore
node_modules
.env
qdrantClien
No results
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
C:\dev\expressjs\candidate-screener-be>curl -X POST -F "cv=@my_cv.pdf" -F "report=@my_report.pdf" http://localhost:3000/upload
{"message":"Files uploaded successfully","cv_id":"cv-1759550668963-131091860.pdf","report_id":"report-1759550668968-145370830.pdf"}
C:\dev\expressjs\candidate-screener-be>curl -X POST -H "Content-Type: application/json" -d "{\"job_title\":\"Backend Developer\",\"cv_id\":\"cv-1759550668963-131091860.pdf\",\"report_id\":\"report-1759550668968-145370830.pdf\"}" "http://localhost:3000/evaluate"
{"job_id":"e1d74f1e-caa0-4152-8689-470a5dc7f7ad","status":"queued"}
C:\dev\expressjs\candidate-screener-be>curl http://localhost:3000/result/e1d74f1e-caa0-4152-8689-470a5dc7f7ad
{"status":"processing","data":{"job_title":"Backend Developer","cv_id":"cv-1759550668963-131091860.pdf","report_id":"report-1759550668968-145370830.pdf"}}
C:\dev\expressjs\candidate-screener-be>curl http://localhost:3000/result/e1d74f1e-caa0-4152-8689-470a5dc7f7ad
{"status":"completed","result":{"cv_match_rate":0.245,"cv_feedback":"The candidate's CV demonstrates strong academic achievements, leadership experience, and professionalism. They have experience in designing .NET microservices and consuming RESTful APIs, showing foundational backend understanding. However, there is a significant mismatch in the core technical skills required for this Backend Developer position (Node.js, Python, SQL, NoSQL, Docker, CI/CD), as their experience primarily revolves around Angular and .NET. Furthermore, the total professional experience is under two years, falling short of the 3-5 years required. A minor typo regarding the current role's end date was also noted.","project_score":2.9,"project_feedback":"The candidate's project report describes a technically sophisticated AI-based candidate screener, demonstrating excellent architectural design, use of advanced concepts like RAG, vector databases (Qdrant), and asynchronous processing (BullMQ with Redis). The report clearly articulates strategies for resilience, scalability, and good documentation. However, the most critical issue is that the submitted project is entirely different from the requested URL shortener service specified in the case study brief. While the reported project showcases impressive skills, it fundamentally failed to address the core requirements of the assigned task.","overall_summary":"Rafli Ramadhani presents as a motivated individual with a strong academic background and impressive leadership skills. Their project report, despite not addressing the requested URL shortener task, demonstrates sophisticated architectural design, an understanding of resilience, and creativity in building a complex AI-based system. However, the candidate's technical experience is predominantly in Angular and .NET, which does not align with the Node.js/Python-centric backend role requirements, and their total professional experience falls short of the desired 3-5 years. The fundamental misinterpretation of the project brief, coupled with the skill and experience mismatch, suggests this candidate is not a suitable fit for this specific Backend Developer position at this time."}}
C:\dev\expressjs\candidate-screener-be>
node
cmd
cmd
OUTLINE
TIMELINE
master*+
Ln 3, Col 15 Spaces: 4 UTF-8 CRLF Ignore Prettier
11:07
04/10/2025


Home Workspaces API Network
Search Postman Ctrl K
Invite
Upgrade
My Workspace
New Import
Overview
GET get
POST New Request
No environment
Collections
Environments
Flows
History
Search collections
Abbauf
API
Belajar
Belajar Strapi
Bookstore
BPJS
INAKA
JOURNEY BANGKIT
mern-app
MORENT
WPU
GET get
GET New Request
GET New Request
GET New Request Copy
GET New Request Copy 2
WPU / get
Save
Share
GET
http://localhost:3000/result/e1d74f1e-caa0-4152-8689-470a5dc7f7ad
Send
Params Authorization Headers (6) Body Scripts Settings
Cookies
Body Cookies Headers (7) Test Results
200 OK 4 ms 2.37 KB
Save Response
JSON
Preview
Visualize
{
"status": "completed",
"result": {
"cv_match_rate": 0.245,
"cv_feedback": "The candidate's CV demonstrates strong academic achievements, leadership experience, and professionalism. They have experience in designing .NET microservices and consuming RESTful APIs, showing foundational backend understanding. However, there is a significant mismatch in the core technical skills required for this Backend Developer position (Node.js, Python, SQL, NoSQL, Docker, CI/CD), as their experience primarily revolves around Angular and .NET. Furthermore, the total professional experience is under two years, falling short of the 3-5 years required. A minor typo regarding the current role's end date was also noted.",
"project_score": 2.9,
"project_feedback": "The candidate's project report describes a technically sophisticated AI-based candidate screener, demonstrating excellent architectural design, use of advanced concepts like RAG, vector databases (Qdrant), and asynchronous processing (BullMQ with Redis). The report clearly articulates strategies for resilience, scalability, and good documentation. However, the most critical issue is that the submitted project is entirely different from the requested URL shortener service specified in the case study brief. While the reported project showcases impressive skills, it fundamentally failed to address the core requirements of the assigned task.",
"overall_summary": "Rafli Ramadhani presents as a motivated individual with a strong academic background and impressive leadership skills. Their project report, despite not addressing the requested URL shortener task, demonstrates sophisticated architectural design, an understanding of resilience, and creativity in building a complex AI-based system. However, the candidate's technical experience is predominantly in Angular and .NET, which does not align with the Node.js/Python-centric backend role requirements, and their total professional experience falls short of the desired 3-5 years. The fundamental misinterpretation of the project brief, coupled with the skill and experience mismatch, suggests this candidate is not a suitable fit for this specific Backend Developer position at this time."
}
}
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
11:08
04/10/2025