

Pemrogramman Web Lanjut

(Aplikasi Web Pembuatan CV - UASCV)



Nama Anggota:

I Nengah Dwi Putra Witarsana	(F1D022049)
Excel Caesariano	(F1D022118)
Ahmad Muslihul Khair	(F1D02310001)
Wicaksono Hadidul Mannan	(F1D02310095)

BAB I

Deskripsi Project

1.1 Latar Belakang

Perkembangan sistem rekrutmen digital telah mendorong meningkatnya kebutuhan akan dokumen Curriculum Vitae (CV) dalam bentuk digital yang terstruktur dan mudah diproses oleh sistem informasi. Dalam praktiknya, sebagian besar perusahaan dan institusi pendidikan kini menggunakan platform berbasis web untuk menerima dan memproses dokumen pelamar. Kondisi ini menuntut pelamar, khususnya mahasiswa dan pencari kerja pemula, untuk memiliki CV yang tidak hanya informatif tetapi juga konsisten, rapi, dan mudah diperbarui.

Namun, proses penyusunan CV masih banyak dilakukan secara manual menggunakan aplikasi pengolah kata atau template statis yang tidak terintegrasi dengan sistem pengelolaan data. Akibatnya, data pribadi, riwayat pendidikan, serta pengalaman kerja sering tersebar di berbagai dokumen terpisah dan harus dimasukkan ulang setiap kali terjadi perubahan atau saat digunakan untuk keperluan yang berbeda. Selain itu, sebagian besar platform pembuat CV daring belum menyediakan mekanisme penyimpanan data pengguna yang aman dan terstruktur, sehingga pengguna berisiko kehilangan data atau mengalami inkonsistensi informasi.

Keterbatasan tersebut menunjukkan adanya kesenjangan antara kebutuhan pengguna terhadap sistem pembuat CV yang terintegrasi dengan ketersediaan solusi yang ada saat ini. Tanpa adanya sistem yang mampu mengelola data pengguna secara terpusat dan menghasilkan CV secara otomatis dalam format digital standar, proses penyusunan CV menjadi tidak efisien dan rawan kesalahan. Kondisi ini berdampak pada kualitas dokumen yang digunakan dalam proses rekrutmen, yang pada akhirnya dapat memengaruhi peluang pelamar.

Berdasarkan permasalahan tersebut, diperlukan sebuah sistem pembuat CV berbasis web yang mampu mengelola data pengguna secara terstruktur, menyediakan autentikasi yang aman, serta menghasilkan dokumen CV dalam format digital yang siap digunakan. Sistem ini diharapkan dapat menjadi solusi terintegrasi yang mendukung proses pembuatan, pembaruan, dan penyajian CV secara lebih efisien dan konsisten.

1.2 Tujuan Project

Adapun tujuan dari proyek ini adalah sebagai berikut:

1. Mengembangkan sistem pembuat Curriculum Vitae berbasis web yang terintegrasi dengan pengelolaan data pengguna untuk menghasilkan CV yang konsisten, terstruktur, dan mudah diperbarui.

2. Menerapkan mekanisme autentikasi dan otorisasi pengguna guna menjamin keamanan, kerahasiaan, dan keandalan data pribadi yang digunakan dalam penyusunan CV.
3. Mengimplementasikan fitur ekspor dokumen CV ke dalam format PDF serta mengevaluasi kinerja sistem berdasarkan kecepatan akses, keakuratan data, dan kemudahan penggunaan.

1.3 Teknologi yang Digunakan

1. Backend

Backend berperan sebagai pusat pengolahan data dan penyedia layanan aplikasi (API).

Teknologi yang digunakan meliputi:

a. Node.js dan Express.js

Node.js digunakan sebagai lingkungan runtime JavaScript di sisi server, sedangkan Express.js berfungsi sebagai framework web untuk membangun restful API. Kombinasi ini memungkinkan sistem menangani permintaan dari frontend, memproses logika bisnis, serta mengelola komunikasi dengan basis data secara efisien.

b. Supabase

Supabase digunakan sebagai layanan backend-as-a-service yang menyediakan basis data relasional berbasis PostgreSQL serta layanan autentikasi pengguna. Supabase memungkinkan penyimpanan data profil pengguna, riwayat pendidikan, dan pengalaman kerja secara terpusat, sekaligus mendukung pengelolaan akun dan keamanan akses.

c. JSON Web Token (JWT)

JWT digunakan untuk mengelola sesi autentikasi pengguna. Setelah pengguna berhasil login, sistem akan menghasilkan token yang digunakan untuk memverifikasi setiap permintaan ke server, sehingga hanya pengguna yang telah terautentikasi yang dapat mengakses dan memodifikasi data CV.

d. Bcrypt

bcrypt digunakan untuk melakukan enkripsi kata sandi sebelum disimpan ke dalam basis data. Dengan demikian, data sensitif pengguna terlindungi dari risiko kebocoran atau penyalahgunaan.

e. Dotenv

dotenv digunakan untuk mengelola variabel lingkungan, seperti kunci akses Supabase dan konfigurasi server, sehingga informasi sensitif tidak dituliskan secara langsung di dalam kode program.

2. Frontend

Frontend berfungsi sebagai antarmuka pengguna untuk mengakses sistem dan mengelola data CV. Teknologi yang digunakan meliputi:

a. Vue.js 3

Vue 3 digunakan sebagai framework JavaScript untuk membangun antarmuka yang reaktif dan modular. Dengan Vue, setiap bagian halaman seperti formulir, tampilan CV, dan navigasi dapat dikelola dalam komponen yang terpisah dan mudah dikembangkan.

b. Vite

Vite digunakan sebagai alat build dan development server yang menyediakan proses pengembangan yang cepat serta efisien, sehingga memudahkan proses pengujian dan pengembangan antarmuka.

c. Vue Router

Vue Router digunakan untuk mengatur navigasi antar halaman, seperti halaman login, dashboard pengguna, dan halaman pembuatan CV.

d. Pinia

Pinia digunakan sebagai state management untuk menyimpan dan mengelola data penting aplikasi, seperti informasi pengguna yang sedang login dan data CV yang sedang diedit.

e. Axios

Axios digunakan sebagai library HTTP client untuk menghubungkan frontend dengan backend melalui API, sehingga data dapat dikirim dan diterima secara real-time.

f. html2pdf.js

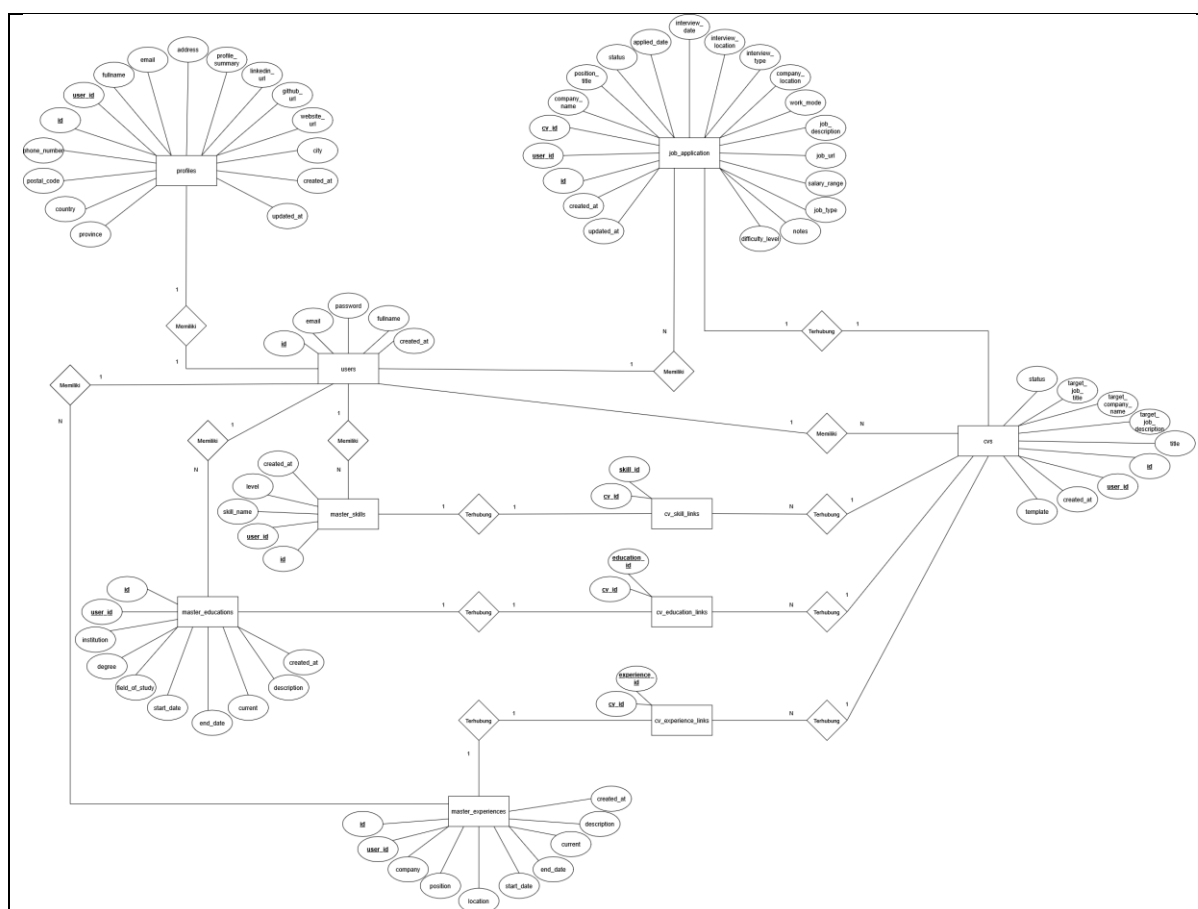
html2pdf.js digunakan untuk mengonversi tampilan CV yang ada di halaman web menjadi dokumen PDF. Fitur ini memungkinkan pengguna mengunduh CV dalam format yang siap digunakan untuk keperluan rekrutmen.

Perancangan Sistem

2.1 Deskripsi Singkat Sistem

UASCV adalah aplikasi pembuatan Curriculum Vitae (CV) berbasis web yang dilengkapi dengan AI untuk membantu pengguna dalam menyusun dan menyesuaikan CV sesuai dengan kebutuhan.

2.2 Desain Database (ERD) dan Penjelasan Singkat



Berdasarkan gambar tersebut terdapat 10 entitas yang memiliki fungsinya masing-masing yaitu:

1. Entitas users yang berfungsi untuk otorisasi pengguna yaitu register dan login sehingga data antar pengguna tidak menyatu serta menjaga privasi pengguna.
2. Entitas profile yang berfungsi sebagai penyedia informasi lebih lanjut dari users, entitas ini akan digunakan dalam pengisian CV.
3. Entitas cvs yang berfungsi sebagai tempat informasi CV pengguna berada.
4. Entitas master_skills berfungsi sebagai tempat keahlian pengguna berada.
5. Entitas master_educations berfungsi sebagai tempat edukasi yang dilalui pengguna berada.

6. Entitas `master_experiences` berfungsi sebagai tempat pengalaman pengguna berada.
7. Entitas `cv_skill_links` berfungsi menghubungkan keahlian dengan CV.
8. Entitas `cv_education_links` berfungsi menghubungkan edukasi yang dilalui dengan CV.
9. Entitas `cv_experience_links` berfungsi menghubungkan pengalaman dengan CV.
10. Entitas `job_application` berfungsi sebagai tempat pengguna mendaftarkan CV ke lokasi kerja.

BAB III

Implementasi

3.1 Implementasi Backend

Implementasi *backend* dibangun menggunakan Node.js dengan *framework* Express.js. *Backend* berfungsi sebagai penyedia REST API yang akan dikonsumsi oleh aplikasi *frontend*.

3.1.1 Struktur Folder

Pengembangan *backend* menggunakan struktur folder MVC (*Model-View-Controller*) yang dimodifikasi untuk memisahkan tanggung jawab kode (*separation of concerns*). Berikut adalah struktur folder utama pada sisi *backend*:

3.1.2 Implementasi API

Aplikasi diinisialisasi pada *file* `app.js`". Pada tahap ini, *server* dikonfigurasi untuk menggunakan *Cross-Origin Resource Sharing* (CORS) agar dapat diakses oleh *frontend*, serta *parsing body request* dalam format JSON.

Endpoint API dikelompokkan berdasarkan modul fungsionalitas dan didaftarkan ke dalam aplikasi Express sebagai berikut:

1. *Auth Route* (`/api/auth`): Menangani registrasi dan *login* pengguna.
2. *Profile Route* (`/api/profiles`): Mengelola data profil pengguna.
3. *CV Route* (`/api/cvs`): Menangani CRUD (*Create, Read, Update, Delete*) data CV.
4. *Master Route* (`/api/master`): Menyediakan data master (referensi).
5. *AI Route* (`/api/ai`): *Endpoint* untuk fitur kecerdasan buatan.
6. *Job Tracker Route* (`/api/job-tracker`): Mengelola pelacakan lamaran kerja.

Contoh implementasi *routing* dapat dilihat pada modul CV (`routes/cvRoute.js`), di mana setiap *endpoint* dipetakan ke fungsi *controller* yang sesuai:

```
// Cuplikan backend/routes/cvRoute.js
import express from 'express';
import { authenticateToken } from '../middleware/authMiddleware.js';
import { getAllCvs, createCv, ... } from '../controllers/cvController.js';

const router = express.Router();

// Endpoint dilindungi oleh middleware authenticateToken
router.get('/', authenticateToken, getAllCvs);
router.post('/', authenticateToken, createCv);
// ... routing lainnya
```

3.1.3 Implementasi Middleware

Untuk menjaga keamanan API, diterapkan *middleware* autentikasi menggunakan JSON Web Token (JWT). *Middleware* ini berfungsi memverifikasi token yang dikirimkan oleh *client* melalui *header Authorization* sebelum mengizinkan akses ke *endpoint* yang dilindungi.

Implementasi `authMiddleware.js` memeriksa keberadaan token dan validitasnya:

```
// Cuplikan backend/middleware/authMiddleware.js
export const authenticateToken = (req, res, next) => {
  const authHeader = req.headers['authorization'];
  const token = authHeader && authHeader.split(" ")[1]; //
  Mengambil token Bearer

  if (!token) return res.status(401).json({ message: "Access
  denied, token required" });

  jwt.verify(token, process.env.JWT_SECRET, (err, user) => {
    if (err) return res.status(403).json({ message: "Invalid
    or expired token" });
    req.user = user;
    next();
  });
};
```

3.2 Implementasi *Frontend*

Implementasi *frontend* dibangun menggunakan *framework* Vue.js 3 dengan dukungan Vite sebagai *build tool* dan Pinia untuk manajemen *state*.

3.2.1 Halaman (*Views*)

Antarmuka pengguna dibagi menjadi beberapa halaman (*Views*) yang merepresentasikan fitur utama aplikasi. Berdasarkan konfigurasi router, halaman-halaman tersebut meliputi:

1. Halaman Publik
 - a. HomeView: Halaman utama (landing page).
 - b. LoginView & RegisterView: Halaman autentikasi pengguna.
2. Halaman Dashboard
 - a. OnboardingView: Halaman awal setelah login.
 - b. JobTrackerView: Halaman fitur pelacakan lamaran kerja (Phase 3).
3. Halaman Editor CV
 - a. Terdiri dari komponen editor terpisah seperti EducationView, ExperienceView, SkillsView, dan SummaryView.

- b. *PreviewView*: Halaman untuk melihat hasil akhir CV dan melakukan ekspor.

3.2.2 Routing

Navigasi antar halaman dikelola menggunakan *Vue Router*. Konfigurasi *router* menerapkan *Navigation Guards* (*router.beforeEach*) untuk membatasi akses ke halaman khusus pengguna yang sudah *login* (*requiresAuth: true*).

Jika pengguna belum *login* mencoba mengakses halaman *dashboard*, sistem akan secara otomatis mengarahkan pengguna kembali ke halaman *login*.

```
// Cuplikan frontend/src/router/index.js
router.beforeEach((to) => {
  document.title = to.meta.title || 'UASCV'

  // Cek apakah route membutuhkan autentikasi
  if (to.meta.requiresAuth && !isLoggedIn.value) {
    return { path: "/auth/login" };
  }
})
```

3.2.3 Konsumsi API

Komunikasi data antara frontend dan backend dilakukan menggunakan library *Axios*. Dibuat sebuah konfigurasi global (*apiClient*) untuk menangani setiap permintaan HTTP. Konfigurasi ini mencakup:

1. *Base URL*: Mengambil alamat API dari *environment variable*.
2. *Request Interceptor*: Secara otomatis menyisipkan token *JWT* dari *localStorage* ke dalam header *Authorization* pada setiap request.
3. *Response Interceptor*: Menangani error secara global, misalnya jika token kadaluwarsa (401 *Unauthorized*), pengguna akan otomatis *logout* dan diarahkan ke halaman *login*.

```
// Cuplikan frontend/src/api/axios.js
apiClient.interceptors.request.use((config) => {
  const token = localStorage.getItem("token");
  if (token) {
    config.headers.Authorization = `Bearer ${token}`;
  }
  return config;
});
```

BAB IV

Pengujian

3.3 Pengujian Postman

API Test Report

26/12/2025, 11.47.04

34
TOTAL

34
PASSED

0
FAILED

POST

Register

201 Created (1173ms)

http://localhost/api/auth/register

REQUEST BODY

RESPONSE BODY

```
{
  "email": "test@example.com",
  "password": "password123",
  "fullname": "Test User"
}
```

```
{
  "message": "Register success"
}
```

ASSERTIONS

✓

Status is 201 or 400/409 handled

POST

Login

200 OK (172ms)

http://localhost/api/auth/login

REQUEST BODY

RESPONSE BODY

```
{
  "email":
  "test@example.com",
  "password":
  "password123"
}
```

```
{
  "message": "Login success",
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6Ijg2c224f9-8f47-4aaa-b171-9020a9005394",
  "user": {
    "id": "86c224f9-8f47-4aaa-b171-9020a9005394",
    "email": "test@example.com",
    "fullname": "Test User"
  }
}
```

ASSERTIONS

✓

Login success or proper error

GET

Get My CVs

http://localhost/api/cvs

200 OK (97ms)

REQUEST BODY

No Body

RESPONSE BODY

```
{
  "data": []
}
```

ASSERTIONS

✓ Status is 200

POST

Create CV

http://localhost/api/cvs

201 Created (448ms)

REQUEST BODY

```
{
  "career_level": "fresh_graduate"
}
```

RESPONSE BODY

```
{
  "message": "CV Created",
  "data": {
    "id": "b783138d-5208-478b-a297-557a572ed433",
    "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
    "title": "Untitled CV",
    "template": "modern",
    "status": "draft",
    "created_at": "2025-12-26T03:46:53.643529",
    "updated_at": "2025-12-26T03:46:53.643529",
    "target_job_title": null,
    "target_company_name": null,
    "target_job_description": null
  }
}
```

ASSERTIONS

✓ Status is 201

PUT

Update CV

http://localhost/api/cvs/b783138d-5208-478b-a297-557a572ed433

200 OK
(111ms)

REQUEST BODY

```
{
  "title": "Updated Title",
  "status": "draft"
}
```

RESPONSE BODY

```
{
  "message": "CV berhasil
diperbarui",
  "data": {
    "id": "b783138d-5208-478b-
a297-557a572ed433",
    "user_id": "86c224f9-8f47-
4aaa-b171-9020a9005394",
    "title": "Updated Title",
    "template": "modern",
    "status": "draft",
    "created_at": "2025-12-
26T03:46:53.643529",
    "updated_at": "2025-12-
26T03:46:53.707",
    "target_job_title": null,
    "target_company_name": null,
    "target_job_description":
null
  }
}
```

GET

Get CV Full Content

http://localhost/api/cvs/b783138d-5208-478b-a297-557a572ed433/full

200 OK
(676ms)

REQUEST BODY

No Body

RESPONSE BODY

```
{
  "cv_info": {
    "id": "b783138d-5208-478b-
a297-557a572ed433",
    "user_id": "86c224f9-8f47-
4aaa-b171-9020a9005394",
    "title": "Updated Title",
    "template": "modern",
    "status": "draft",
    "created_at": "2025-12-
26T03:46:53.643529",
    "updated_at": "2025-12-
26T03:46:53.707",
    "target_job_title": null,
    "target_company_name": null,
    "target_job_description":
null
  },
  "sections": {

```

<div><div>POST</div><div><div>Toggle Item In CV</div><div>http://localhost/api/cvs/b783138d-5208-478b-a297-557a572ed433/toggle</div></div></div> <div>200 OK (250ms)</div>	
REQUEST BODY	RESPONSE BODY

```
{
  "section": "educations",
  "itemId": ""
}
```

```
{
  "message": "Item added to CV",
  "status": "added"
}
```

<div><div>GET</div><div><div>List Educations</div><div>http://localhost/api/master/educations</div></div></div> <div>200 OK (81ms)</div>	
REQUEST BODY	RESPONSE BODY
No Body	<pre>{ "data": [] }</pre>
ASSERTIONS	
✓ Status code is 200	

<div><div>POST</div><div><div>Add Education</div><div>http://localhost/api/master/educations</div></div></div> <div>201 Created (123ms)</div>	
REQUEST BODY	RESPONSE BODY
<pre>{ "institution": "University X", "start_date": "2020-01-01", "end_date": "2024-01-01", "current": false }</pre>	<pre>{ "message": "Item Saved", "data": { "id": "9350a7bb-ea0e-4c0d-8752-4721b9aeaf54", "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394", "institution": "University X", "degree": null, "field_of_study": null, "start_date": "2020-01-01", "end_date": "2024-01-01", "current": false, "description": null, "created_at": "2025-12-26T03:46:55.879396" } }</pre>
ASSERTIONS	
✓ Status is 201	

PUT

Update Education

http://localhost/api/master/educations/9350a7bb-ea0e-4c0d-8752-4721b9aeaf54

200 OK
(113ms)

REQUEST BODY

```
{
  "institution": "University X
(Updated)"
}
```

RESPONSE BODY

```
{
  "message": "Item Updated",
  "data": {
    "id": "9350a7bb-ea0e-4c0d-8752-4721b9aeaf54",
    "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
    "institution": "University X
(Updated)",
    "degree": null,
    "field_of_study": null,
    "start_date": "2020-01-01",
    "end_date": "2024-01-01",
    "current": false,
    "description": null,
    "created_at": "2025-12-26T03:46:55.879396"
  }
}
```

ASSERTIONS

✓

Status is 200

DELETE

Delete Education

http://localhost/api/master/educations/9350a7bb-ea0e-4c0d-8752-4721b9aeaf54

200 OK
(81ms)

REQUEST BODY

No Body

RESPONSE BODY

```
{
  "message": "Item Deleted"
}
```

ASSERTIONS

✓

Status is 200

GET	List Experiences http://localhost/api/master/experiences	200 OK (93ms)
------------	--	---------------

REQUEST BODY	RESPONSE BODY
No Body	<pre>{ "data": [] }</pre>
ASSERTIONS	
<div>✓ Status code is 200</div>	

POST	Add Experience http://localhost/api/master/experiences	201 Created (102ms)
REQUEST BODY	RESPONSE BODY	
<pre>{ "company": "Company X", "position": "Developer", "start_date": "2021-01-01", "end_date": "2023-01-01" }</pre>	<pre>{ "message": "Item Saved", "data": { "id": "970d1167-77bc-46f6-bc9b-af6fdb127f2", "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394", "company": "Company X", "position": "Developer", "location": null, "start_date": "2021-01-01", "end_date": "2023-01-01", "current": false, "description": null, "created_at": "2025-12-26T03:46:57.077941" } }</pre>	
ASSERTIONS		
✓ Status is 201		

PUT

Update Experience

http://localhost/api/master/experiences/970d1167-77bc-46f6-bc9b-af6fdb127f2

200 OK
(102ms)

REQUEST BODY

```
{  
  "position": "Senior Developer"  
}
```

RESPONSE BODY

```
{  
  "message": "Item Updated",  
  "data": {
```

```
    }  
  }  
}
```

```
    "id": "970d1167-77bc-46f6-bc9b-af6fdb127f2",  
    "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",  
    "company": "Company X",  
    "position": "Senior Developer",  
    "location": null,  
    "start_date": "2021-01-01",  
    "end_date": "2023-01-01",  
    "current": false,  
    "description": null,  
    "created_at": "2025-12-26T03:46:57.077941"  
  }  
}
```

ASSERTIONS

✓

Status is 200

DELETE

Delete Experience

http://localhost/api/master/experiences/970d1167-77bc-46f6-bc9b-af6fdb127f2

200 OK
(99ms)

REQUEST BODY

No Body

RESPONSE BODY

```
{  
  "message": "Item Deleted"  
}
```

ASSERTIONS

✓

Status is 200

GET List Skills http://localhost/api/master/skills		200 OK (97ms)
REQUEST BODY		RESPONSE BODY
No Body		<pre>{ "data": [] }</pre>
ASSERTIONS		

✓ Status code is 200

POST Add Skill http://localhost/api/master/skills		201 Created (109ms)
REQUEST BODY		RESPONSE BODY
<pre>{ "skill_name": "JavaScript", "level": "Intermediate" }</pre>		<pre>{ "message": "Item Saved", "data": { "id": "f3c30f15-c6eb-4b4d-81e2-7f7c3874941d", "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394", "skill_name": "JavaScript", "level": "Intermediate", "created_at": "2025-12-26T03:46:58.258985" } }</pre>
ASSERTIONS		
✓ Status is 201		

<div> <div>PUT</div> <div> <div>Update Skill</div> <div>http://localhost/api/master/skills/f3c30f15-c6eb-4b4d-81e2-7f7c3874941d</div> </div> <div>200 OK (119ms)</div> </div>	
<div>REQUEST BODY</div> <div> <pre>{ "level": "Advanced" }</pre> </div>	<div>RESPONSE BODY</div> <div> <pre>{ "message": "Item Updated", "data": { "id": "f3c30f15-c6eb-4b4d-81e2-7f7c3874941d", "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394", "skill_name": "JavaScript", "level": "Advanced", "created_at": "2025-12-26T03:46:58.258985" } }</pre> </div>
<div>ASSERTIONS</div>	

✓ Status is 200

<div> <div>DELETE</div> <div> <div>Delete Skill</div> <div>http://localhost/api/master/skills/f3c30f15-c6eb-4b4d-81e2-7f7c3874941d</div> </div> <div>200 OK (87ms)</div> </div>	
<div>REQUEST BODY</div> <div>No Body</div>	<div>RESPONSE BODY</div> <div> <pre>{ "message": "Item Deleted" }</pre> </div>
<div>ASSERTIONS</div> <div>✓ Status is 200</div>	

<div> <div>GET</div> <div> <div>Get My Profile</div> <div>http://localhost/api/profiles</div> </div> <div>200 OK (98ms)</div> </div>	
<div>REQUEST BODY</div> <div>No Body</div>	<div>RESPONSE BODY</div> <div> <pre>{ "data": {} }</pre> </div>

<div>PUTSave Profilehttp://localhost/api/profiles200 OK (113ms)</div>	
REQUEST BODY	RESPONSE BODY
<pre>{ "fullname": "Test User", "email": "test@example.com", "phone_number": "+62123456789", "city": "City", "country": "Indonesia", "profile_summary": "Short bio here.", "linkedin_url": "https://linkedin.com/in/test", "github_url": "https://github.com/test"}</pre>	<pre>{ "message": "Profil berhasil disimpan", "data": { "id": "c9a2e545-f2c1-4e5f-9fb1-833bc789583d", "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394", "fullname": "Test User", "email": "test@example.com", "address": null, "profile_summary": "Short bio here.", "linkedin_url":</pre>

<pre> "https://linkedin.com/in/test", "github_url": "https://github.com/test", "website_url": null,</pre>

<div>POSTGenerate Experience Descriptionhttp://localhost/api/ai/experience500 Internal Server Error (619ms)</div>	
REQUEST BODY	RESPONSE BODY
<pre>{ "position": "Software Engineer", "company": "Company ABC", "keywords": "Node.js, Express, SQL"}</pre>	<pre>{ "message": "Gagal generate deskripsi", "error": "{\n \"error\": {\n \"code\": 429,\n \"message\": \"You exceeded your current quota, please check your plan and billing details.\\n\\nFor more information on this error, head to: https://ai.google.dev/gemini-api/docs/quotas\\n\\n* Quota exceeded for metrics: 20, model: gemini-2.5-flash\\n\\nPlease retry in 23.437426ms.\\n\\nstatus\": \"quota_exceeded\", \"url\": \"https://ai.google.dev/gemini-api/docs/rate-limit-quotas\", \"url\": \"https://ai.google.dev/gemini-api/docs/rate-limit-quotas\", \"type\": \"type.googleapis.com/google.rpc.Help\", \"links\": [{\n \"type\": \"type.googleapis.com/google.rpc.QuotaFailure\", \"violation\": \"quotaMetric\", \"generativelanguage.googleapis.com/generate_content: FreeTier\", \"quotaDimensions\": {\n \"location\": \"global\", \"model\": \"gemini-2.5-flash\", \"type\": \"type.googleapis.com/google.rpc.RetryInfo\", \"retryDelay\": 23.437426ms\n }\n }\n }\"}</pre>

<div>POSTGenerate Professional Summaryhttp://localhost/api/ai/summary500 Internal Server Error (333ms)</div>	
REQUEST BODY	RESPONSE BODY
<pre>{ "fullname": "Test User", "jobTitle": "Frontend Developer", "keywords": "Vue.js, Tailwind, UI", "experience_level": "Menengah"}</pre>	<pre>{ "message": "Gagal generate summary", "error": "{\n \"error\": {\n \"code\": 429,\n \"message\": \"You exceeded your current quota, please check your plan and billing details.\\n\\nFor more information on this error, head to: https://ai.google.dev/gemini-api/docs/quotas\\n\\n* Quota exceeded for metrics: 20, model: gemini-2.5-flash\\n\\nPlease retry in 59.527515641ms.\\n\\nstatus\": \"quota_exceeded\", \"url\": \"https://ai.google.dev/gemini-api/docs/rate-limit-quotas\", \"url\": \"https://ai.google.dev/gemini-api/docs/rate-limit-quotas\", \"type\": \"type.googleapis.com/google.rpc.Help\", \"links\": [{\n \"type\": \"type.googleapis.com/google.rpc.QuotaFailure\", \"violation\": \"quotaMetric\", \"generativelanguage.googleapis.com/generate_content: FreeTier\", \"quotaDimensions\": {\n \"location\": \"global\", \"model\": \"gemini-2.5-flash\", \"type\": \"type.googleapis.com/google.rpc.RetryInfo\", \"retryDelay\": 59.527515641ms\n }\n }\n }\"}</pre>

POST

Generate Style Config

http://localhost/api/ai/style

500 Internal Server Error (371ms)

REQUEST BODY

```
{
  "prompt": "Saya ingin tampilan CV yang modern, elegan, berwarna biru tua, cocok untuk melamar ke
```

RESPONSE BODY

```
{
  "message": "Gagal generate style",
  "error": "{\n  \"error\": {\n    \"code\": 429,\n    \"message\": \"You exceeded your information on this error, head to: https://ai.google.dev/gemini-api/https://ai.dev/usage?tab=rate-limit. \\n* Quota exceeded for metric 20, model: gemini-2.5-flash\\nPlease retry in 59.003920232s.\\",\n    \"status\": {\n      \"@type\": \"type.googleapis.com/google.rpc.Help\", \"links\": [{\n        \"description\": \"Quota exceeded for metric 20, model: gemini-2.5-flash\\nPlease retry in 59.003920232s.\\",\n        \"url\": \"https://ai.google.dev/gemini-api/docs/rate-limit\" \n      }]\n    },\n    \"quotaMetric\": \"generativelanguage.googleapis.com/generate_content_free_quota\", \"quotaDimensions\": {\n      \"location\": \"global\", \"model\": \"gemini-2.5-flash\" \n    }\n  },\n  \"retryDelay\": \"59.003920232s\" \n}"
```

POST

Create CV Helper

http://localhost/api/cvs

201 Created (100ms)

REQUEST BODY

```
{
  "title": "CV for Job Tracker",
  "career_level": "mid_level"
}
```

RESPONSE BODY

```
{
  "message": "CV Created",
  "data": {
    "id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
    "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
    "title": "CV for Job Tracker",
    "template": "modern",
    "status": "draft",
    "created_at": "2025-12-26T03:47:01.502994",
    "updated_at": "2025-12-26T03:47:01.502994",
    "target_job_title": null,
    "target_company_name": null,
    "target_job_description": null
  }
}
```

POST

Create Job Application

http://localhost/api/job-tracker

201 Created (113ms)

REQUEST BODY

```
{
  "company_name": "Google",
  "position_title": "Software Engineer",
  "status": "wishlist",
  "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
  "job_description": "Descriptive text",
  "company_location": "Jakarta",
  "difficulty_level": "medium"
}
```

RESPONSE BODY

```
{
  "id": "69eaed4d-8dd4-43b9-970b-c295a7df79c5",
  "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
  "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
  "company_name": "Google",
  "position_title": "Software Engineer",
  "status": "wishlist",
  "applied_date": null,
  "interview_date": null,
  "interview_location": null,
  "interview_type": null,
  "company_location": "Jakarta",
  "work_mode": null,
  "job_description": "Descriptive text"
}
```

ASSERTIONS

✓ Status code is 201

✓ Response has created application

GET

Get All Job Applications

http://localhost/api/job-tracker

200 OK (91ms)

REQUEST BODY

No Body

RESPONSE BODY

```
[
  {
    "id": "69eaed4d-8dd4-43b9-970b-c295a7df79c5",
    "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
    "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
    "company_name": "Google",
    "position_title": "Software Engineer",
    "status": "wishlist",
    "applied_date": null,

```

RESPONSE BODY

```
    "interview_date": null,
    "interview_location": null,
    "interview_type": null,
    "company_location": "Jakarta",

```

ASSERTIONS

✓ Status code is 200

✓ Response is an array

GET

Get Single Job Application

http://localhost/api/job-tracker/69eaed4d-8dd4-43b9-970b-c295a7df79c5

200 OK
(93ms)

REQUEST BODY

No Body

RESPONSE BODY

```
{
  "id": "69eaed4d-8dd4-43b9-970b-c295a7df79c5",
  "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
  "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
  "company_name": "Google",
  "position_title": "Software Engineer",
  "status": "wishlist",
  "applied_date": null,
  "interview_date": null,
  "interview_location": null,
  "interview_type": null,
  "company_location": "Jakarta",
  "work_mode": null,
  "job_description": "Descriptive text"
}
```

ASSERTIONS

✓ Status code is 200

✓ Response has required fields

PUT

Update Job Application

http://localhost/api/job-tracker/69eaed4d-8dd4-43b9-970b-c295a7df79c5

200 OK
(388ms)

REQUEST BODY

RESPONSE BODY

```
{
  "company_name": "Google Updated",
  "status": "applied"
}
```

```
{
  "id": "69eaed4d-8dd4-43b9-970b-c295a7df79c5",
  "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
  "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
  "company_name": "Google Updated",
  "position_title": "Software Engineer",
  "status": "applied",
  "applied_date": null,
  "interview_date": null,
  "interview_location": null,
  "interview_type": null,
  "company_location": "Jakarta",
  "work_mode": null,
  "job_description": "Descriptive text"
}
```

ASSERTIONS

✓ Status code is 200

✓ Application updated

PATCH

Update Job Application Status

http://localhost/api/job-tracker/69eaed4d-8dd4-43b9-970b-c295a7df79c5/status

200 OK
(304ms)

REQUEST BODY

```
{
  "status": "interview"
}
```

RESPONSE BODY

```
{
  "id": "69eaed4d-8dd4-43b9-970b-c295a7df79c5",
  "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
  "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
  "company_name": "Google Updated",
  "position_title": "Software Engineer",
  "status": "interview",
  "applied_date": null,
  "interview_date": null,
  "interview_location": null,
  "interview_type": null,
  "company_location": "Jakarta",
  "work_mode": null,
  "job_description": null
}
```

ASSERTIONS

✓ Status code is 200

✓ Status updated

GET

Get Job Applications by Status

http://localhost/api/job-tracker/by-status

200 OK (356ms)

REQUEST BODY

No Body

RESPONSE BODY

```
{
  "wishlist": [],
  "applied": [],
  "interview": [
    {
      "id": "69eaed4d-8dd4-43b9-970b-c295a7df79c5",
      "user_id": "86c224f9-8f47-4aaa-b171-9020a9005394",
      "cv_id": "e7e8310a-ecad-48e8-b7dd-dfad476115f8",
      "company_name": "Google Updated",
      "position_title": "Software Engineer",
      "status": "interview",
      "applied_date": null,
      "interview_date": null,
      "interview_location": null
    }
  ]
}
```

ASSERTIONS

✓ Status code is 200

✓ Response has status groups

GET Get Job Application Stats 200 OK (104ms)	
http://localhost/api/job-tracker/stats	
REQUEST BODY	RESPONSE BODY
No Body	<pre>{ "wishlist": 0, "applied": 0, "interview": 1, "offer": 0, "total": 1 }</pre>
ASSERTIONS	

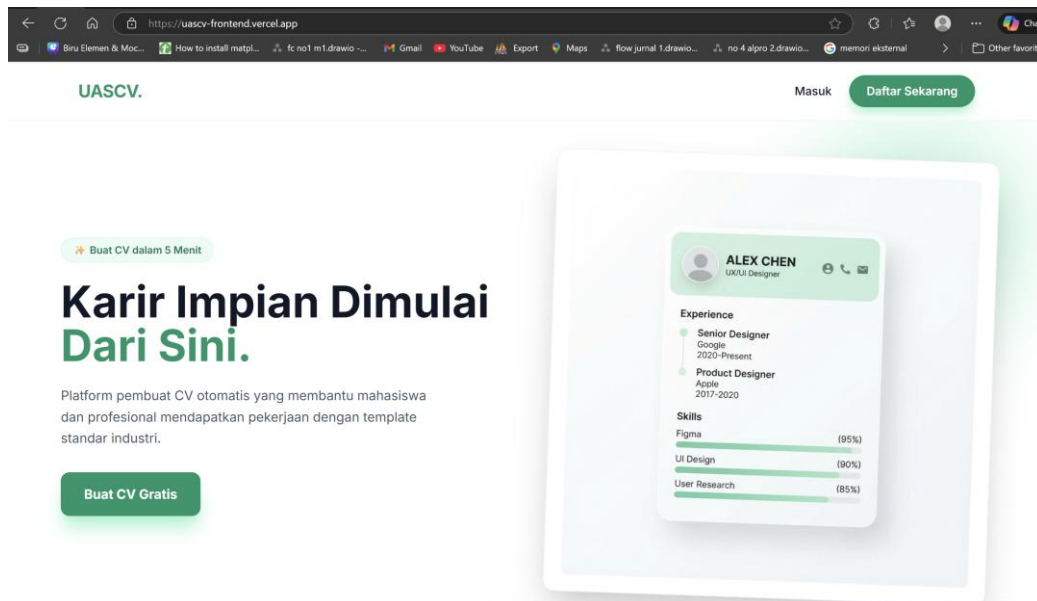
- ✓ Status code is 200
- ✓ Response has stats

DELETE Delete Job Application 200 OK (175ms)	
http://localhost/api/job-tracker/69eaed4d-8dd4-43b9-970b-c295a7df79c5	
REQUEST BODY	RESPONSE BODY
No Body	<pre>{ "message": "Job application deleted successfully" }</pre>
ASSERTIONS	
<ul style="list-style-type: none"> ✓ Status code is 200 ✓ Deletion confirmed 	

DELETE Delete CV 200 OK (109ms)	
http://localhost/api/cvs/e7e8310a-ecad-48e8-b7dd-dfad476115f8	
REQUEST BODY	RESPONSE BODY
No Body	<pre>{ "message": "CV berhasil dihapus" }</pre>
ASSERTIONS	
<ul style="list-style-type: none"> ✓ Status is 200 	

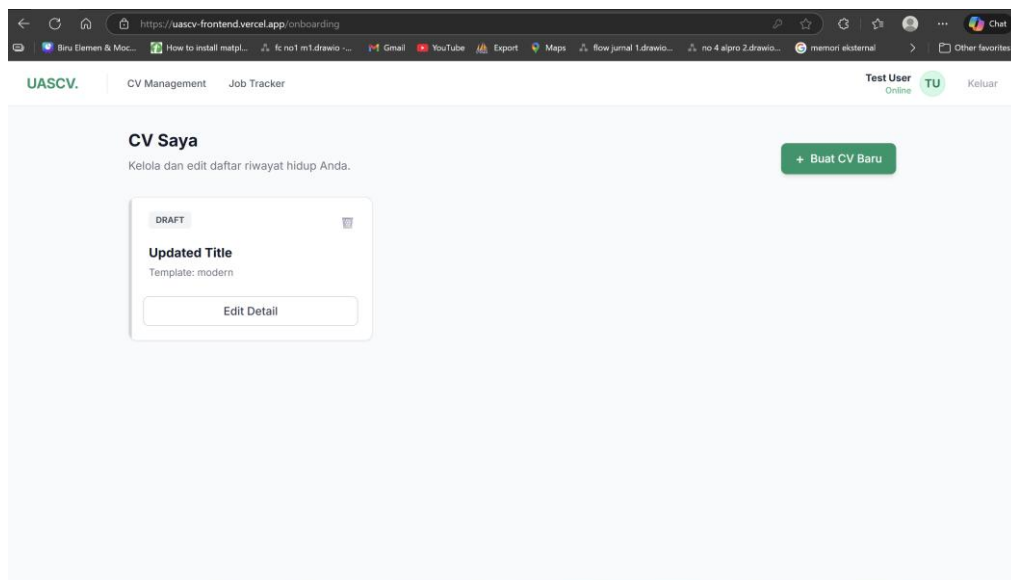
3.4 Pengujian Aplikasi

1. Halaman Utama (Landing Page)



Halaman depan yang menyambut pengguna dengan desain modern dan informatif. Menampilkan ringkasan fitur unggulan aplikasi seperti pembuatan CV dengan AI dan berbagai pilihan template menarik. Terdapat tombol akses cepat untuk mendaftar atau masuk ke dalam akun pengguna.

2. Onboarding & Login



Halaman utama setelah login berfungsi sebagai pusat kontrol pengguna. Di sini, Anda dapat melihat daftar riwayat hidup (CV) yang telah dibuat, status terakhirnya, dan tombol untuk membuat CV baru. Halaman ini dirancang agar pengguna dapat dengan mudah mengelola banyak versi CV untuk berbagai lamaran kerja.

3. Informasi Pribadi

JUDUL DOKUMEN

Updated Title

Status: Draft

Target Lamaran

KONTEKS AI

Tips: Isi bagian ini agar fitur **AI Generator** di menu Pengalaman & Ringkasan bisa menyesuaikan isinya secara spesifik dengan lowongan yang Anda incar.

POSISI YANG DILAMAR

Contoh: Senior Backend Engineer

NAMA PERUSAHAAN TUJUAN

Contoh: GoTo, Traveloka, Google

DESKRIPSI / KUALIFIKASI LOWONGAN

Paste kualifikasi pekerjaan dari job portal di sini... (AI akan menggunakan ini untuk mencocokkan skill Anda)

Kontak & Profil

Sync Master Data

Nama Lengkap

Test User

Email

test@example.com

No. Handphone

+62123456789

LinkedIn URL

https://linkedin.com/in/test

Alamat Lengkap

Kota

City

Provinsi

Negara

Indonesia

Kode Pos

Ringkasan Profil (Master)

Short bio here.

*Ringkasan ini bersifat umum. Gunakan menu "Ringkasan" di sidebar untuk ringkasan spesifik per CV.

Formulir pengisian data diri utama yang menjadi fondasi dari setiap CV. Pengguna dapat melengkapi informasi vital seperti Nama Lengkap, Email, Nomor Telepon, dan Tautan LinkedIn. Data ini dapat disinkronisasi dari Master Data sehingga tidak perlu diisi ulang untuk setiap CV baru.

4. Edit Pendidikan

The screenshot shows the 'Tambah Pendidikan' (Add Education) form in the UASCV application. The form is a modal window with the following fields:

- Institusi ***: Text input field.
- Jenjang**: Dropdown menu with 'Sarjana (S1)' selected.
- Jurusan**: Text input field.
- Mulai ***: Date input field (dd/mm/yyyy).
- Selesai**: Date input field (dd/mm/yyyy).
- ☐ **Masih sekolah di sini**: Checkbox.
- Batal** and **Simpan Baru** buttons at the bottom right.

Bagian ini memungkinkan pengguna untuk menambahkan latar belakang pendidikan formal maupun non-formal. Formulir mencakup detail institusi, gelar, jurusan, serta periode waktu pendidikan. Pengguna dapat menambahkan lebih dari satu riwayat pendidikan sesuai kebutuhan.

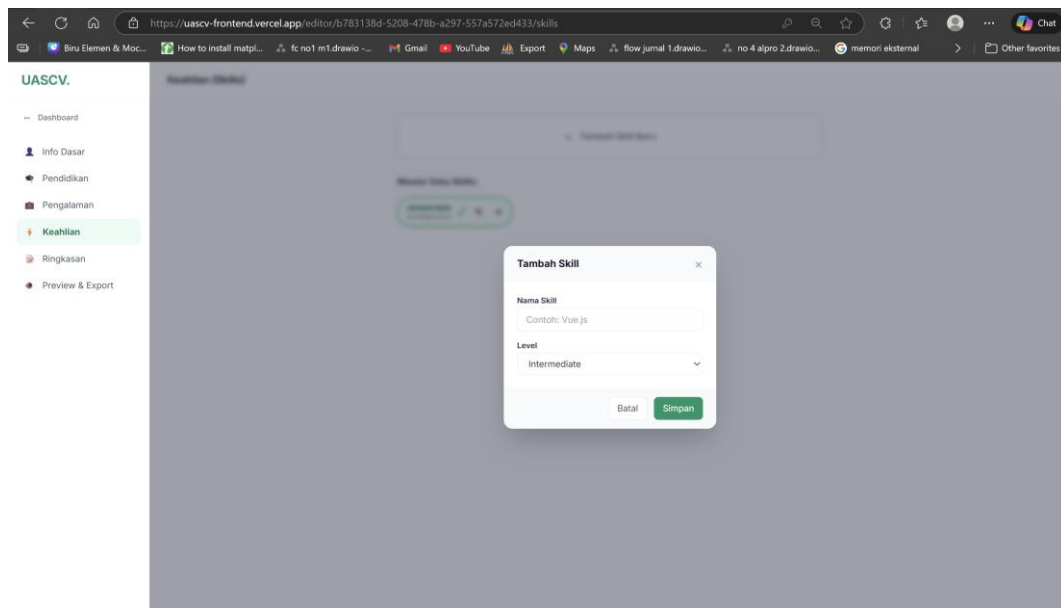
5. Edit Pengalaman Kerja

The screenshot shows the 'Tambah Pengalaman Baru' (Add New Experience) form in the UASCV application. The form is a modal window with the following fields:

- Posisi ***: Text input field.
- Perusahaan ***: Text input field.
- Lokasi**: Text input field.
- Mulai ***: Date input field (dd/mm/yyyy).
- Selesai**: Date input field (dd/mm/yyyy).
- ☐ **Masih bekerja di sini**: Checkbox.
- AI GENERATOR** section: Includes a text area for keywords (e.g., 'Vue.js, Team Lead, Optimasi...'), a 'Generate' button, and a note: '* Tips: Isi "Target Pekerjaan" di menu Info Dasar agar hasil AI lebih spesifik & relevan.' Below this is a text area for 'Deskripsi Pekerjaan' with a note: 'Hasil generate AI akan muncul di sini...'
- Batal** and **Simpan Baru** buttons at the bottom right.

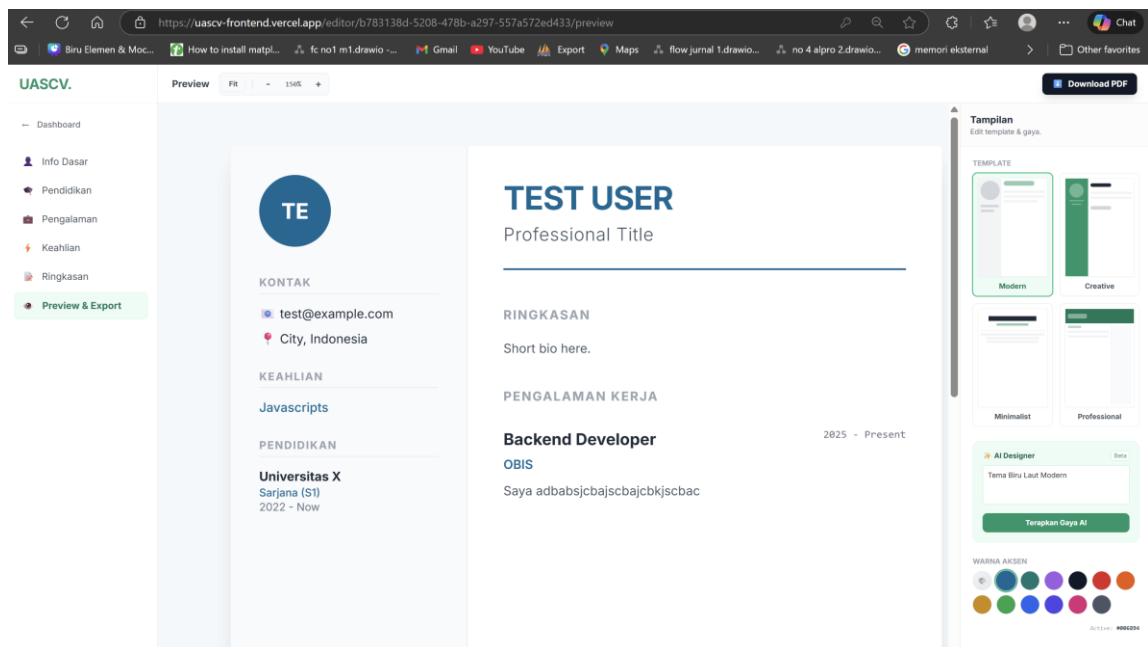
Fitur unggulan di mana pengguna dapat mendeskripsikan pengalaman kerja mereka. Dilengkapi dengan AI Generator yang dapat membantu merangkai kalimat deskripsi pekerjaan yang profesional berdasarkan posisi dan nama perusahaan. Memudahkan pengguna untuk menonjolkan pencapaian mereka dengan bahasa yang standar industri.

6. Edit Keahlian (Skills)



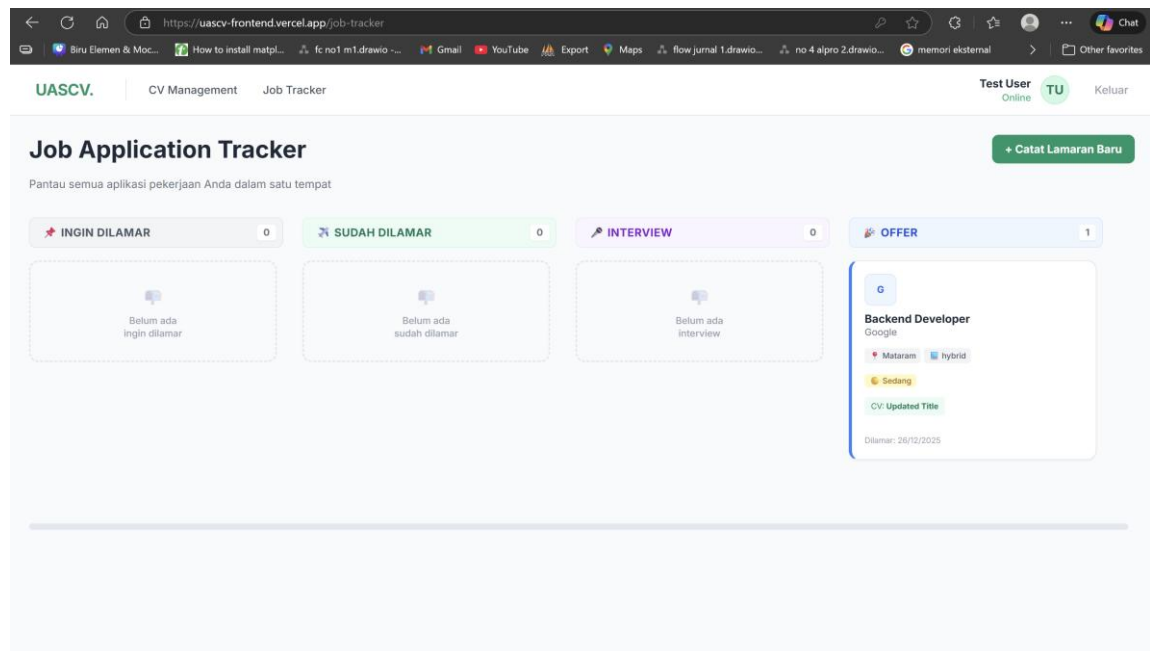
Halaman manajemen keahlian untuk menampilkan kompetensi teknis maupun soft skills. Pengguna dapat menentukan tingkat kemahiran (Level) untuk setiap skill, mulai dari Pemula hingga Ahli. Tampilan visual yang jelas membantu perekrut memahami kualifikasi kandidat dengan cepat.

7. Preview CV



Tampilan akhir CV yang mensimulasikan hasil cetak dalam format A4. Di sini pengguna bisa melihat penerapan template (Modern, Creative, dll) secara real-time. Jika sudah sesuai, pengguna dapat langsung mengunduh file CV dalam format PDF berkualitas tinggi siap kirim.

8. Job Tracker



Fitur tambahan berbentuk papan Kanban untuk manajemen proses pelamaran kerja. Pengguna dapat memindahkan kartu lamaran antar status: *Wishlist*, *Applied*, *Interview*, hingga *Offer*. Sangat berguna untuk memantau progres karir agar terorganisir dengan rapi.

LAMPIRAN

Link Github: <https://github.com/cuikww/uascv-project>

Link Deployment : <https://uascv-frontend.vercel.app>