BASE URL

https://etbackend-v2-usy9.onrender.com

All paths below are relative to this.

AUTH / USERS

Prefix: /api/auth/

Get CSRF token (MANDATORY FIRST CALL)

GET /api/auth/csrf/

Response

{ "csrftoken": "<token>" }

- Also sets csrftoken cookie

- Frontend must send X-CSRFToken on all write requests

- Always use credentials: "include"

Signup (CONFIRMED)

POST /api/auth/signup/

Payload (JSON – EXACT)

{

  "email": "user@example.com",

  "password": "StrongPass@123",

  "confirm_password": "StrongPass@123",

  "newsletter": true,

  "invite_code": "OPTIONAL"

}

 confirm_password **IS REQUIRED**
invite_code optional

 Do NOT send name fields here

Verify Email

POST /api/auth/verify-email/

```
{

  "email": "user@example.com",

  "code": "123456"

}
```

Resend Verification Email

POST /api/auth/resend-verification/

```
{

  "email": "user@example.com"

}
```

Login (HttpOnly JWT – CANONICAL)

POST /api/auth/token/

```
{

  "email": "user@example.com",

  "password": "StrongPass@123"

}
```

Result

- access_token → HttpOnly cookie
- refresh_token → HttpOnly cookie
- Frontend stores **nothing**

Refresh Token

POST /api/auth/token/refresh/

No payload (uses cookie)

Logout

POST /api/auth/logout/

No payload

Change Password

POST /api/auth/change-password/

```
{
  "old_password": "OldPass@123",
  "new_password": "NewPass@123"
}
```

Add Feedback

POST /api/auth/add-feedback/

multipart/form-data

urgency=HIGH

subject=Bug Report

message=Resume parsing failed

emoji= 🙄

attachment=<optional file>

Current User

GET /api/auth/users/me/

No payload

ORGANIZATION

Prefix: /api/organization/

Create Organization (CONFIRMED)

POST /api/organization/organizations/

multipart/form-data

name=EnabledTalent

industry=1

employee_size=2

headquarter_location=Toronto

about=AI hiring platform

linkedin_url=https://linkedin.com/company/enabledtalent

url=https://enabledtalent.com

avatar=<optional file>

industry & employee_size are **INTEGER CHOICES**
Do NOT send users / root_user / ids

List Organizations

GET /api/organization/organizations/

Update Organization

PATCH /api/organization/organizations/{id}/

Same fields as create (partial allowed)

Create Organization Invite

POST /api/organization/organizations/{id}/create-invite/

{

  "email": "invitee@example.com"

}

CANDIDATES

Prefix: /api/candidates/

Create Candidate Profile (CONFIRMED)

POST /api/candidates/profiles/

multipart/form-data

resume_file=<optional pdf/docx>

video_pitch=<optional file>

willing_to_relocate=true

employment_type_preferences=["Full-time","Internship"]

work_mode_preferences=["Remote","Hybrid"]

has_workvisa=false

expected_salary_range=80k-100k

is_available=true

disability_categories=["Vision"]

accommodation_needs=YES

workplace_accommodations=["Screen reader"]

✓ All fields optional
✓ Profile auto-linked to logged-in user

Get Candidate Profile

GET /api/candidates/profiles/{slug}/

Upload / Replace Resume

PATCH /api/candidates/profiles/{slug}/

multipart/form-data

resume_file=<pdf or docx>

Parse Resume (Celery – ASYNC)

POST /api/candidates/profiles/{slug}/parse-resume/

No payload
Triggers background job

Parsing Status

GET /api/candidates/profiles/{slug}/parsing-status/

Verify Profile (User-edited confirmation)

POST /api/candidates/profiles/{slug}/verify-profile/

{

  "name": "Jane Doe",

  "email": "jane@example.com",

```
  "linkedin": "https://linkedin.com/in/jane",

  "skills": ["Python","NLP"],

  "work_experience": [

    { "company": "X", "role": "Y" }

  ]

}
```

Notes

POST /api/candidates/notes/

multipart/form-data

identifier=resume

note=Strong ML background

section=Experience

selected_text=Built OCR pipeline

context={"page":2}

note_file=<optional>

Resume Prompt Chat

POST /api/candidates/prompt/

```
{

  "input_text": "Summarize my resume",

  "resume_slug": "john-abc123",

  "thread_id": null

}
```

Career Coach

POST /api/candidates/career-coach/

```
{

  "input_text": "How can I improve my resume?",

  "resume_slug": "john-abc123",

  "thread_id": null
```

}

JOBS / AI

Prefix: /api/channels/

Create Job

POST /api/channels/jobs/

```
{
  "title": "ML Engineer",
  "job_desc": "Python + NLP",
  "workplace_type": "Remote",
  "location": "Toronto",
  "job_type": "Full-time",
  "skills": ["Python","NLP"],
  "estimated_salary": "100k-130k",
  "visa_required": false
}
```

Rank Candidates (ASYNC)

POST /api/channels/jobs/{job_id}/rank-candidates/

No payload

Get Ranking Results

GET /api/channels/jobs/{job_id}/ranking-data/

AI Agent

POST /api/channels/agent/

```
{
  "query": "Find best ML candidates"
}
```

One-line Flow Summary

Signup → Verify → Login (cookies) → Create profile → Upload resume → Parse (Celery) → Review → Rank → AI insights