ChatGPT

Africa Global Talent Platform – API Specification (MVP)

This document defines RESTful API contracts for the MVP. Each endpoint lists the HTTP method, path, request and response body, status codes, and required authentication/authorization. All responses are JSON and include a `success` boolean and either `data` or `error` fields.

1. Authentication

Register

Attribute	Description
Method	`POST`
Path	`/auth/register`
Request Body	`{ "email": "user@example.com", "password": "string", "role": "candidate"	"employer" }`
Response	`201 Created` – `{ "success": true, "data": { "id": 1, "email": "user@example.com", "role": "candidate", "token": "<jwt>" } }` `400 Bad Request` – missing/invalid fields `409 Conflict` – email already exists
Auth & AuthZ	Public (no token). Role must be `candidate` or `employer`; admin accounts are created manually.

Login

Attribute	Description
Method	`POST`
Path	`/auth/login`
Request Body	`{ "email": "user@example.com", "password": "string" }`
Response	`200 OK` – `{ "success": true, "data": { "id": 1, "email": "user@example.com", "role": "candidate", "token": "<jwt>" } }` `401 Unauthorized` – invalid credentials
Auth & AuthZ	Public (no token). Returns JWT token on success.

Get Current User

Attribute	Description
Method	`GET`
Path	`/auth/me`
Response	`200 OK` – `{ "success": true, "data": { "id": 1, "email": "user@example.com", "role": "candidate" } }` `401 Unauthorized` – missing/invalid token
Auth & AuthZ	Requires valid JWT. Returns minimal user information.

2. Candidate Profile

Create or Update Profile

Attribute	Description
Method	`PUT`
Path	`/candidates/me`
Request Body	`{ "fullName": "Jane Doe", "country": "KEN", "summary": "Experienced software engineer...", "skills": ["JavaScript", "React"], "experienceYears": 4, "education": "B.Sc. Computer Science" }`

Response	`200 OK` – `{ "success": true, "data": { ...updated candidate profile... } }`
	 `400 Bad Request` – invalid fields
	 `401 Unauthorized` – not authenticated
	 `403 Forbidden` – user role not `candidate`

Auth & AuthZ	Only authenticated users with role `candidate`.

Get My Profile

Attribute	Description
Method	`GET`
Path	`/candidates/me`

Response	`200 OK` – `{ "success": true, "data": { ...candidate profile or null... } }`
	 `401 Unauthorized` – not authenticated
	 `403 Forbidden` – user role not `candidate`

Auth & AuthZ	Only authenticated users with role `candidate`.

Upload Resume

Attribute	Description
Method	`POST`
Path	`/candidates/me/resume`
Request	`multipart/form-data` with a file field named `resume` (PDF).
Response	`201 Created` – { "success": true, "data": { "resumeUrl": "https://..." } } `400 Bad Request` – invalid file type or size `401 Unauthorized` – not authenticated `403 Forbidden` – user role not `candidate`
Auth & AuthZ	Only authenticated candidates. File is uploaded to object storage; `resumeUrl` is a pre-signed link.

3. Employer / Company

Create or Update Company Profile

Attribute	Description
Method	`PUT`
Path	`/companies/me`
Request Body	{ "name": "Acme Inc.", "industry": "Software", "country": "USA", "description": "Global tech company", "websiteUrl": "https://acme.com" }
Response	`200 OK` – { "success": true, "data": { ...company profile... } } `400 Bad Request` – invalid fields `401 Unauthorized` – not authenticated `403 Forbidden` – user role not `employer`
Auth & AuthZ	Only authenticated users with role `employer`. Admin can update any company via a separate admin endpoint if needed.

Get My Company Profile

Attribute	Description
Method	`GET`
Path	`/companies/me`
Response	`200 OK` – { "success": true, "data": { ...company profile or null... } } `401 Unauthorized` – not authenticated `403 Forbidden` – user role not `employer`

Attribute	Description
Auth & AuthZ	Only authenticated employers.

4. Jobs

Create Job Posting

Attribute	Description
Method	`POST`
Path	`/jobs`
Request Body	`{ "title": "Software Engineer", "description": "Detailed description...", "requiredSkills": ["JavaScript", "Node.js"], "country": "USA"	null, "eligibleCountries": ["KEN", "NGA"] (optional), "isRemote": true/false, "visaSponsorship": true/false, "relocationAssistance": true/false, "category": "Engineering", "applicationDeadline": "2026-03-31T23:59:59Z" }`

Response	`201 Created` – `{ "success": true, "data": { "id": 42, ...job... } }`
	 `400 Bad Request` – missing required fields
	 `401 Unauthorized` – not authenticated
	 `403 Forbidden` – user role not `employer`

Auth & AuthZ	Only authenticated employers may create jobs. Jobs are created with `status='pending'` and require admin approval.

List Jobs

Attribute	Description
Method	`GET`
Path	`/jobs`
Query Parameters	`q` – keyword search; `country` – ISO code; `remote` – boolean; `visa` – boolean; `relocation` – boolean; `category` – string; `page` – integer; `pageSize` – integer
Response	`200 OK` – `{ "success": true, "data": { "items": [ { ...job summary... } ], "page": 1, "pageSize": 10, "total": 123 } }`
Auth & AuthZ	Public. Only jobs with `status='active'` are returned.

Get Job Details

Attribute	Description
Method	`GET`
Path	`/jobs/{jobId}`
Response	`200 OK` – `{ "success": true, "data": { ...job details including eligibility flags... } }` `404 Not Found` – job does not exist or is not active/pending and user is not authorised
Auth & AuthZ	Public for active jobs; employers can view their own pending jobs; admins can view any.

Update Job (Employer)

Attribute	Description
Method	`PUT`
Path	`/jobs/{jobId}`
Request Body	Same as create job, but all fields optional.

Response	`200 OK` – `{ "success": true, "data": { ...updated job... } }`
	 `400 Bad Request`
	 `401 Unauthorized`
	 `403 Forbidden` – employer does not own job or job already approved
	 `404 Not Found`

Auth & AuthZ	Only employers who created the job and only while `status='pending'`. Admins can update via admin endpoints.

Close Job (Employer)

Attribute	Description
Method	`PATCH`
Path	`/jobs/{jobId}/close`

Response	`200 OK` – `{ "success": true, "data": { "status": "closed" } }`
	 `401 Unauthorized`
	 `403 Forbidden` – employer does not own job
	 `404 Not Found`

Auth & AuthZ	Employers or admins may close an active or pending job they own.

5. Applications

Apply to a Job

Attribute	Description
Method	`POST`
Path	`/jobs/{jobId}/applications`
Request Body	`{ "coverLetter": "I am excited to apply..." }`

Response	`201 Created` – `{ "success": true, "data": { "id": 101, "status": "applied" } }`
	 `400 Bad Request` – invalid job or missing resume/profile
	 `401 Unauthorized`
	 `403 Forbidden` – user role not `candidate`
	 `404 Not Found` – job not found or not active
	 `409 Conflict` – duplicate application

Auth & AuthZ	Only authenticated candidates. Candidates must have a complete profile and resume.

List Applications for a Job (Employer)

Attribute	Description
Method	`GET`
Path	`/jobs/{jobId}/applications`

Response	`200 OK` – `{ "success": true, "data": { "items": [ { "id": 101, "candidate": { ...profile summary... }, "status": "applied", "appliedAt": "..." } ] } }`
	 `401 Unauthorized`
	 `403 Forbidden` – employer does not own job and user is not admin
	 `404 Not Found`

Auth & AuthZ	Employers can view applications for their own jobs; admins can view any. Candidates cannot access this endpoint.

List My Applications (Candidate)

Attribute	Description
Method	`GET`
Path	`/applications/me`

Response	`200 OK` – `{ "success": true, "data": { "items": [ { "id": 101, "job": { "id": 42, "title": "Software Engineer", "companyName": "Acme" }, "status": "under_review", "appliedAt": "...", "updatedAt": "..." } ] } }`
	 `401 Unauthorized`
	 `403 Forbidden` – user role not `candidate`

Attribute	Description
Auth & AuthZ	Candidates see only their own applications.

Update Application Status (Employer/Admin)

Attribute	Description
Method	`PATCH`
Path	`/applications/{applicationId}`
Request Body	`{ "status": "under_review"	"shortlisted"	"hired"	"rejected" }`

Response	`200 OK` – `{ "success": true, "data": { "id": 101, "status": "shortlisted" } }`
	 `400 Bad Request` – invalid status or transition
	 `401 Unauthorized`
	 `403 Forbidden` – employer does not own associated job or user not admin
	 `404 Not Found`

Auth & AuthZ	Employers may update status for applications to their jobs; admins may update any. Candidates cannot update application status.

6. Admin Moderation

Admin endpoints require the user to have the `admin` role.

List Pending Jobs

Attribute	Description
Method	`GET`
Path	`/admin/jobs/pending`
Response	`200 OK` – `{ "success": true, "data": { "items": [ { ...job... } ] } }`
Auth & AuthZ	Admin only.

Approve Job

Attribute	Description
Method	`PATCH`
Path	`/admin/jobs/{jobId}/approve`

Attribute	Description

Response	`200 OK` – `{ "success": true, "data": { "id": 42, "status": "active" } }`
	 `404 Not Found` – job does not exist or already active
	 `409 Conflict` – job status is not `pending`

Auth & AuthZ	Admin only.

Reject Job

Attribute	Description
Method	`PATCH`
Path	`/admin/jobs/{jobId}/reject`
Response	`200 OK` – `{ "success": true, "data": { "id": 42, "status": "closed" } }`
Auth & AuthZ	Admin only. Setting status to `closed` marks the job as not available.

List Users

Attribute	Description
Method	`GET`
Path	`/admin/users`
Response	`200 OK` – `{ "success": true, "data": { "items": [ { "id": 1, "email": "...", "role": "candidate", "active": true } ] } }`
Auth & AuthZ	Admin only. Supports query parameters such as `role`, `status` for filtering.

Deactivate User

Attribute	Description
Method	`PATCH`
Path	`/admin/users/{userId}/deactivate`
Response	`200 OK` – `{ "success": true, "data": { "id": 1, "active": false } }`
Auth & AuthZ	Admin only. Prevents the user from logging in and marks associated profiles and jobs as closed.

7. Application Lifecycle

Applications move through the following states:

State	Description	Next States
`applied`	Candidate submitted the application. Default initial state.	`under_review`, `rejected`
`under_review`	Employer opened the application and is evaluating.	`shortlisted`, `rejected`
`shortlisted`	Employer likes the candidate and may interview or do further checks.	`hired`, `rejected`
`hired`	Candidate accepted the job offer. Terminal state.	–
`rejected`	Candidate was not selected. Terminal state.	–

Invalid transitions (e.g., moving directly from `applied` to `hired`) should return `400 Bad Request`. Application state changes must be performed by the employer or admin.

8. Role-based Access Summary

Role	Permissions
Candidate	Register & login; create/update own profile; upload resume; search and view jobs; apply to jobs; view own applications and statuses.
Employer	Register & login; create/update company profile; post jobs; view/edit own pending jobs; close own jobs; view applications for their jobs; update application statuses (within valid transitions).
Admin	Manage users (list, deactivate); approve or reject jobs; view all jobs and applications; update any job or application; view/edit any company profile; moderate content.

All endpoints return standard HTTP status codes and a JSON body. Clients should include the JWT token in the `Authorization: Bearer <token>` header. CSRF protection can be handled via HTTP-only cookies for web clients.