



Delivery Execution Plan for Africa Global Talent Platform

This plan moves the product from its current designed state to a live, trusted and usable platform. The process follows six sequential phases. At the end of each phase, acceptance criteria and deliverables are outlined, along with checkpoints and any identified blockers.

Phase 1 – MVP Execution & Completion

Objectives

Turn designs and scaffolding into a fully working MVP without changing scope or adding new features.

MVP Acceptance Criteria

Candidate Flow - Candidate can register (`POST /auth/register` with role = `candidate`) and log in (`POST /auth/login`). - Candidate can complete their profile (`PUT /candidates/me`) and upload a resume via pre-signed upload URL (`POST /candidates/me/resume/upload-url` → `PUT` → `POST /candidates/me/resume`). - Candidate can search jobs using keywords and filters (country, remote, visa sponsorship, relocation, category) via `GET /jobs`. - Candidate can view individual job details via `GET /jobs/{jobId}`. - Candidate can apply for an active job via `POST /jobs/{jobId}/applications` only after completing their profile and uploading a resume. - Candidate can view their application history and statuses via `GET /applications/me`. - Candidate receives transactional emails upon application submission and when an employer updates the status (basic notification system).

Employer Flow - Employer can register (`POST /auth/register` with role = `employer`) and log in. - Employer can create/update their company profile (`PUT /companies/me`) and upload a logo via pre-signed URL. - Employer can create a job (`POST /jobs`), providing title, description, skills, location/remote flags, visa/relocation flags, category and deadline. New jobs are `pending` until an admin approves. - Employer can view a list of their jobs (pending, active, closed) and see the number of applications per job. - Employer can view applications for their jobs (`GET /jobs/{jobId}/applications`); access candidate profiles and resumes via pre-signed GET URLs. - Employer can update application statuses (`PATCH /applications/{applicationId}`) following valid transitions (applied → under_review → shortlisted → hired/rejected).

Admin Flow - Admin can log in using their credentials (role = `admin`). - Admin has an overview dashboard showing counts of pending jobs and users. - Admin can view and moderate pending jobs (`GET /admin/jobs/pending`), approving or rejecting them (`PATCH /admin/jobs/{id}/approve` or `/reject`). Approved jobs become active; rejected jobs become closed. - Admin can list users (`GET /admin/users`) and deactivate accounts (`PATCH /admin/users/{id}/deactivate`). - Admin can manage static content

(FAQ, visa guides, etc.) via a simple content management interface (e.g., editing markdown or database entries).

Remaining Implementation Gaps

Backend - Implement the missing API endpoints described in the spec: - `PUT /jobs/{jobId}` for employers to edit pending jobs. - `PATCH /jobs/{jobId}/close` for employers/admins to close jobs. - `GET /jobs?mine=true` or a dedicated `/employers/me/jobs` endpoint to list an employer's jobs. - `GET /jobs/{jobId}/applications` and `GET /applications/me` as per spec. - `PATCH /applications/{applicationId}` to update application status with lifecycle validation. - Admin endpoints for job approval/rejection and user management. - Pre-signed URL generation endpoints for resumes and logos. - Email notification service integration (e.g., using AWS SES or SendGrid). - Search indexing service integration to push jobs to the search index on create/update and remove on close. - Validation and error handling middleware (using Joi or express-validator). - Logging and basic monitoring.

Frontend - Build all pages mapped in the UX plan: landing pages, job listing/search, job detail, sign-up/login forms, candidate profile form, resume upload flow, application form, dashboards (candidate, employer, admin), employer job posting form, application review page, admin moderation panels, and resource pages. - Implement state management and API integration with JWT handling. - Implement loading, empty, and error states in UI components. - Handle permissions and route guards based on roles (candidate/employer/admin). - Connect search UI to the backend search endpoint with filters.

DevOps - Finish database migrations and apply them across environments (dev/staging/prod). - Provision search engine (e.g., Elasticsearch) and connect the backend. Configure backups. - Provision object storage (e.g., S3) and configure IAM policies. - Configure email sending service (SES/SendGrid) and store credentials securely. - Complete CI/CD pipelines defined earlier: build, test, lint; deploy to dev and staging. Add database migration step to pipeline. - Set up monitoring and alerting (basic logs, error tracking, health checks).

Prioritized Execution Checklist

1. **Backend Core Endpoints** – finish remaining CRUD and admin endpoints; add role-based checks; integrate validation and error handling. (High priority)
2. **File Upload & Storage** – implement pre-signed URL endpoints and integrate S3; update database accordingly. (High priority)
3. **Email Notifications** – integrate email service and send notifications on application submission and status changes. (High priority)
4. **Search Index Integration** – implement search service wrapper; index jobs on create/update and remove on close. (Medium priority)
5. **Frontend Pages** – develop critical pages in this order: candidate signup/login → profile completion → job search → job detail → application form → candidate dashboard; then employer signup → company profile → job posting → employer dashboard → application review; finally, admin dashboard and moderation tools. (High to medium priority)
6. **Role-Based Routing & Guards** – implement frontend route protection and proper redirection for unauthorized users. (High priority)

7. **DevOps & Environment Setup** – finalise database migrations, search and storage provisioning; configure CI/CD pipelines; set environment variables and secrets; deploy to a staging environment. (Medium priority)
8. **Testing** – create unit and integration tests for backend endpoints and frontend components; run tests in CI pipeline. (Medium priority)

Definition of Done

A feature or flow is considered “DONE” when:

- It meets all acceptance criteria and has no critical bugs.
- It is covered by automated tests where feasible.
- It is integrated into the CI/CD pipeline and deploys successfully to staging.
- It has been reviewed (code review + basic QA) and documented (API/usage notes).
- For UI components, loading, error and empty states are implemented.

Phase 1 Checkpoint: □ MVP functionality clearly defined and executable.

Blockers: None identified beyond implementation effort and coordination among teams.

Phase 2 – Internal Testing & Dogfooding

Objectives

Use the product internally to catch issues before external users do, focusing on core flows and trust.

Dogfooding Scenarios

1. Candidate Happy Path:

2. Create a candidate account, complete the profile and upload a resume.
3. Search for jobs with various filters; view job details.
4. Apply to a job successfully; verify confirmation email.
5. Check candidate dashboard for status updates.

6. Candidate Error Paths:

7. Attempt to apply without completing the profile → expect a helpful error message.
8. Attempt to apply twice to the same job → expect a duplicate application error.
9. Search with filters that yield no results → see an informative empty state.

10. Employer Happy Path:

11. Create an employer account, complete company profile and upload logo.
12. Post a job and confirm it shows in their dashboard as pending.
13. After admin approves the job, verify it appears on the public job listing.
14. View applications, check candidate details, change statuses and confirm notifications are sent.

15. Employer Error Paths:

16. Attempt to post a job without required fields → validation error.

17. Attempt to update an already approved job → expect forbidden.

18. **Admin Happy Path:**

19. Admin logs in; sees pending jobs count.

20. Reviews a pending job; approves or rejects; verify status transitions.

21. Deactivates a user and confirms they cannot log in.

22. **Permission Boundaries:**

23. Candidate cannot access `/jobs/{id}/applications` or `/admin` endpoints.

24. Employer cannot view candidate applications for jobs they do not own.

25. Admin can perform all actions but sees audit logs.

Dogfooding Checklist

- [] All happy path scenarios complete without errors.
- [] Error messages are clear and actionable.
- [] Loading indicators and empty states appear appropriately.
- [] Role-based access works as intended; unauthorized access returns 403.
- [] Email notifications arrive timely for submission and status changes.
- [] Search results and filters behave as expected; no unexpected zero results for common queries.
- [] Resumes and logos upload and download securely via pre-signed URLs.
- [] Job approval workflow works end-to-end (pending → active → closed).

Bug & Friction Report

During dogfooding, track all bugs and frictions. Categorise by severity: - **Critical** – prevents core flow completion (e.g., cannot apply to jobs, search broken). - **Major** – degrades user experience significantly (e.g., invalid error messages, broken filters). - **Minor** – cosmetic or low impact (e.g., UI alignment issues, typos).

Rank fixes according to severity and effort. Address critical and major issues before moving to the pilot.

Phase 2 Checkpoint: ☐ Core flows work without confusion or trust gaps.

Blockers: None if dogfooding feedback is addressed; issues identified will become tasks before the pilot.

Phase 3 – Closed Pilot with Real Users

Objectives

Validate the platform's usefulness and trustworthiness with a small group of real users.

Pilot Design

- **Participants:** 10–30 candidates from Africa and the diaspora (via targeted outreach to professional networks and community groups) and 3–5 employers who are explicitly open to African talent and have real job openings.
- **Duration:** 4–6 weeks or until at least 20 applications are submitted.
- **Onboarding:** Send personalised invitations explaining the pilot scope. Provide a short training session or guide. Set expectations about response times and feedback participation.

Success Metrics

Metric	Target
Candidate sign-up completion	≥ 80 % of invited candidates create accounts and complete profiles.
Job posting completion	All pilot employers post at least one job.
Application completion	≥ 60 % of posted jobs receive at least one application.
Employer response rate	≥ 80 % of applications receive a status update within 7 days.
Time to first response	≤ 72 hours for employers to move applications to <code>under_review</code> .
Candidate satisfaction	Average rating ≥ 4/5 in pilot survey.
Employer satisfaction	Average rating ≥ 4/5 in pilot survey.

Feedback Collection

- **Surveys:** Short forms at the midpoint and end of the pilot covering ease of use, perceived quality of jobs, and overall satisfaction.
- **Structured Interviews:** 1:1 sessions with a subset of candidates and employers to probe deeper into friction points, trust issues and potential improvements.
- **Behavioural Signals:** Analytics on search queries, filter usage, time spent on pages, drop-off points and response times.

Phase 3 Checkpoint: ☐ Real users complete real hiring flows.

Blockers: Any critical issues identified in the pilot that cannot be resolved quickly will delay broader launch.

Phase 4 – Trust, Safety & Compliance

Objectives

Ensure the platform protects candidates and employers and complies with relevant regulations.

Trust Rules

- **Job Eligibility:** Only list jobs that explicitly accept applications from African countries or the diaspora. Jobs must include at least one of: remote eligibility, visa sponsorship, relocation assistance or be based in an African country.

- **Employer Verification:** Employers must verify their identity and company legitimacy (e.g., business registration documents or LinkedIn page). Jobs from unverified employers are not published.
- **Scam Prevention:** Prohibit postings that request payment or personal documents beyond standard resumes. Automatically flag suspicious language (e.g., “fees required”).

Policy Drafts

- **Terms of Service:** Cover platform usage, intellectual property, liability limits, and dispute resolution. Specify that employers are responsible for job accuracy and fairness, and candidates own their data and can request deletion.
- **Privacy Policy:** Describe data collected (profiles, resumes, usage logs), storage practices (encrypted at rest, controlled access), and sharing policies (only with employers reviewing applications). Include cookie usage and user rights (GDPR-style rights to access, rectify, delete).
- **Job Posting Guidelines:** Outline eligibility criteria, prohibited content (discrimination, scams, unrealistic claims), required fields and moderation process. Include consequences for violating guidelines.

Moderation Workflows

1. **Submission Review:** All new job postings default to pending. Admins review within 24 hours, checking eligibility and content.
2. **Reporting:** Users can report job posts or employers via a “Report” button. Reports go to an admin queue with details (user, job, reason). Admins investigate and take action (remove post, warn user, deactivate account).
3. **Enforcement:** Three-strike policy for employers and candidates who violate terms. Serious offences (fraudulent jobs, harassment) lead to immediate deactivation.
4. **Audit Logging:** All moderation actions (approvals, rejections, deactivations) are logged with timestamps and admin IDs.

Phase 4 Checkpoint: Platform is safe, transparent, and defensible.

Blockers: Legal review of policies and potential jurisdiction-specific requirements.

Phase 5 – Soft Launch & Distribution

Objectives

Introduce the product cautiously, build credibility and seed growth.

Soft Launch Strategy

- **Communities:** Announce in targeted online communities (LinkedIn groups, diaspora networks, tech hubs) focusing on African professionals seeking global work. Use authentic messaging emphasising Africa-inclusive jobs.
- **Partnerships:** Partner with African professional organisations, diaspora associations and ethical recruitment agencies to source candidates and reputable employers. Offer co-branding opportunities.
- **Founder-Led Channels:** Leverage founders’ personal networks and social media to tell the story behind the platform and invite early adopters.

- **Email Marketing:** Send curated job digests to pilot participants and waitlist sign-ups. Include success stories from the pilot to build trust.

Messaging Framework

- **What It Is:** A recruiting platform dedicated to connecting African and diaspora talent with global opportunities that are remote, visa-sponsored or relocation-friendly.
- **What It Is Not:** Not a get-rich-quick scheme, not limited to local jobs only, not an agency taking commission from candidates.
- **Value Proposition:** High-quality, vetted jobs; inclusive and fair hiring; equal access to global opportunities; clear eligibility criteria.
- **Tone:** Professional, empowering and transparent. Avoid hype; focus on mission and real stories.

Early Growth Hypotheses

1. **Employer → Candidate Loop:** When employers find qualified candidates, they post more jobs and refer other employers. Measuring job repost rate and employer referrals will indicate this loop's strength.
2. **Candidate → Employer Loop:** Satisfied candidates will refer peers and share on social media. Measuring sign-up sources and referral codes will reveal the loop's effectiveness.
3. **Content → Engagement Loop:** Publishing useful resources (visa guides, career tips) will attract organic traffic and convert readers into candidates or employers. Monitor resource page views and subsequent sign-ups.

Phase 5 Checkpoint: ☐ Users arrive with correct expectations and trust.

Blockers: None if soft launch messaging and community engagement are executed as planned.

Phase 6 – AI Readiness & Roadmap

Objectives

Prepare the platform to adopt AI responsibly once sufficient data is available.

Potential AI Use Cases

1. **Resume Parsing:** Automatically extract skills, experience and education from uploaded resumes to populate candidate profiles and improve search/matching accuracy.
2. **Matching Improvements:** Use machine learning to learn weights for skills and experiences based on successful hires, enhancing the rule-based matching algorithm.
3. **Job Recommendations:** Recommend jobs to candidates based on their profile, search history and similar users' behaviour. Provide explainable reasons for each recommendation.

Data Requirements

- A large corpus of structured candidate profiles (fields + parsed resumes) and job descriptions.
- Historical application outcomes (status transitions, hires) to train matching models.
- Interaction data: search queries, clicks, time spent on listings and acceptance/rejection decisions.
- Explicit user feedback on recommendations to evaluate and adjust model performance.

Guardrails & Risk Mitigation

- **Bias Mitigation:** Exclude protected characteristics (race, ethnicity, nationality, religion, gender, disability) from model features. Use fairness metrics (e.g., equal opportunity difference) to monitor models. Periodically audit recommendations across different groups.
- **Explainability:** Use transparent models (e.g., gradient-boosted decision trees with SHAP values) and surface the top factors driving each recommendation to users and employers.
- **Human Override:** Keep recommendation systems advisory. Candidates and employers can see and override AI suggestions. Do not auto-reject or auto-approve candidates without human review.
- **Privacy:** Anonymise data used for training; obtain user consent for data usage; allow opt-out.

Phased AI Rollout

1. **Data Collection Phase** (pilot + early launch): Instrument data collection with user consent; store structured and interaction data.
2. **Pilot AI Tools Internally:** Build a resume parser and test it on a subset of resumes; compare extracted data with manual profiles.
3. **Introduce AI Suggestions (Beta):** Offer job recommendations behind a feature flag; gather feedback on relevance and fairness.
4. **Expand Matching Algorithms:** Incorporate machine-learning-based scoring into the existing rule-based engine; show a match score breakdown including AI contributions.
5. **Review & Iterate:** Continuously monitor fairness metrics and user feedback; adjust models and retraining frequency accordingly.

Phase 6 Checkpoint: ☐ Platform is ready for responsible AI adoption.

Blockers: Data volume and quality; obtaining legal advice for AI usage policies.

Final Summary & Next 90-Day Priorities

Completed Phases

- **Phase 1:** MVP acceptance criteria defined; engineering checklists created; definition of done set.
- **Phase 2:** Dogfooding scenarios and testing checklist devised; bug report framework established.
- **Phase 3:** Closed pilot plan developed with success metrics and feedback mechanisms.
- **Phase 4:** Trust and safety framework drafted, including terms, privacy policy, posting guidelines and moderation playbook.
- **Phase 5:** Soft launch strategy and messaging framework defined; early growth hypotheses identified.
- **Phase 6:** AI readiness roadmap drafted with data requirements and guardrails.

Remaining Risks

- **Technical Debt:** Rushing to deliver the MVP may leave gaps in validation, error handling and tests. Allocate time for refactoring and addressing technical debt before scale.
- **Trust & Safety:** Malicious actors may attempt to exploit the platform; the moderation process needs adequate staffing and tooling.
- **User Adoption:** If initial employers or candidates do not engage, network effects may stall. Mitigate via continuous outreach and incentives.

- **Data Quality:** Poorly structured job descriptions or resumes will hamper future AI models.
Encourage clear job postings and guide candidates on resume formatting.
- **Regulatory Compliance:** Data protection and labour laws vary across jurisdictions. Obtain legal review for terms, privacy and AI usage before expansion.

Next 90-Day Priorities

1. **Complete MVP Development and QA** – finish all remaining backend, frontend and DevOps tasks; run dogfooding and fix critical issues.
2. **Run Closed Pilot** – onboard pilot users, monitor metrics and collect feedback; iterate quickly on friction points.
3. **Finalise Trust & Safety Policies** – refine and publish Terms of Service, Privacy Policy and Guidelines; train moderation staff.
4. **Deploy Soft Launch** – execute community and partnership outreach; monitor growth loops and adjust messaging.
5. **Instrument Data Collection** – start collecting structured and interaction data responsibly to prepare for AI phase.
6. **Plan Fundraising / Partnerships** – secure resources for hiring and infrastructure to support scaling and AI development.

This phased approach ensures a candidate-first, Africa-inclusive, explainable and trustworthy platform that can scale responsibly without major rewrites.
