**ATS Phase 1 Blueprint**

**Goal:** Launch an **AI-native Applicant Tracking System** with a single “killer wedge” feature that instantly makes recruiters feel like they’re using something from the future.

**1. Core Vision for Phase 1**

We won’t try to compete on *everything* — we’ll dominate in **resume parsing + contextual AI matching** while keeping the rest minimal but functional.  
This way:

* Development cost & time stay low.
* Early users experience something no other ATS does.
* We can layer features later without redoing the core.

**2. Target User & Market Entry**

**Primary target for launch:**

* **Tech startups & SMBs** with <100 employees hiring across multiple roles.
* Recruiters/agencies serving these companies.

**Why?**

* They’re frustrated with clunky big-name ATS tools.
* They’re open to AI-first workflows.
* They care about speed over bureaucracy.

**3. Core Phase 1 Features**

**Must-Have (Launch Ready)**

1. **AI Resume Parsing Engine**
   * Accepts PDF, DOCX, image scans, LinkedIn profile URLs.
   * Extracts **skills, experience, education, certifications, portfolio links**.
   * Uses GPT-based contextual understanding (e.g., “Java” → programming vs coffee).
2. **Contextual Job Matching**
   * Matches candidates to jobs not just on keyword overlap, but **skill depth, industry relevance, and role seniority**.
   * Assigns a **Fit Score** with explainability (“Matched because…”).
3. **Lightweight Pipeline View**
   * Kanban-style drag & drop for candidates across stages: *Applied → Shortlist → Interview → Offer → Hired*.
4. **Basic Job Posting**
   * Create job posts and share as a **public job page** with a shortlink.
   * Auto-generate description drafts from a role title.
5. **Email Integration (Basic)**
   * Send interview invites or rejections directly from the ATS.

**Nice-to-Have (Optional for Phase 1.5)**

* Auto-interview scheduling with Google/Outlook calendar.
* Candidate nurturing sequences.
* Blind-screening toggle for bias elimination.

**4. Technical Architecture**

**Frontend**

* **Framework:** Next.js (React 19 for SSR + speed)
* **UI Library:** TailwindCSS + ShadCN/UI for rapid, modern UI components
* **State Management:** Zustand or Redux Toolkit

**Backend**

* **Core API:** Node.js (Express or NestJS)
* **Database:** PostgreSQL (Supabase for faster dev)
* **AI Layer:**
  + GPT-4o / Claude 3.5 Sonnet for parsing & matching logic.
  + Embedding models (OpenAI text-embedding-3-large) for semantic skill matching.

**Integrations**

* **File handling:** AWS S3 or Supabase storage.
* **Email:** Gmail API (OAuth2) for free-tier, switch to transactional email (Postmark/SendGrid) for scaling.
* **Job Board Posting (later):** LinkedIn API, Indeed API.

**5. AI Logic Flow**

**Resume Parsing:**

Upload Resume → OCR if needed → AI extraction → Store structured JSON:

{

name, contact, skills: [{ skill, confidence }], experience: [...],

education: [...], portfolio\_links: [...]

}

**Job Matching:**

Candidate profile + Job description → Convert both to embeddings →

Cosine similarity scoring → Top matches → AI-generated Fit Score explanation

**6. Development Phases**

**Phase 1 (4–6 weeks)**

* Core parsing engine with GPT + embeddings.
* Minimal UI for uploading resumes and viewing parsed data.
* Kanban pipeline.
* Basic job posting and email integration.

**Phase 1.5 (2–3 weeks after launch)**

* Improved parsing accuracy.
* Blind-screening mode.
* Integration with LinkedIn for profile importing.

**7. Market Entry Strategy**

* **Private beta** with 10–20 SMB recruiters.
* Offer **free tier** for 1–2 roles at a time.
* Run **case studies** showing “Time-to-shortlist reduced by 80%”.
* Build credibility → Expand features → Charge per seat or per job slot.

**8. CRM Expansion Path**

Once ATS works:

1. Convert **candidate database** into **contact database**.
2. Add **client-side project pipelines** alongside hiring pipelines.
3. Expand integrations into sales, support, and marketing automation → the CRM revolution begins.