

UX Research: LinkedIn & Job Platform Analysis

Scope: Job Boards, Candidate Matching Platforms, and Candidate-Facing ATS Portals

Platforms Reviewed

Job Boards & Applications	LinkedIn, Indeed, Glassdoor, Reed, Totaljobs
Candidate-Role Matching Platforms	Hired, Otta, Wellfound (AngelList Talent), Triplebyte, Cord, Talent.io, Honeypot
ATS/Recruiting Tools (Candidate View)	Workday, Greenhouse, Lever

1) Job Application Flows & Friction Points

Common Patterns

- Multi-stage flows: Filter → Job list → Job detail → Apply.
- Multiple application entry points (list card, detail page, recommendations).
- Optional pre-screening questions and employer-specific fields.

Friction Points

- External ATS redirects break user context and increase drop-off.
- Required fields without guidance lead to abandonment.
- Login/sign-up gates appear before value is demonstrated.
- Resume upload can overwrite or reset previously entered data.

Recommendations

- Show time-to-apply estimate and steps upfront (e.g., “~5 min, 3 steps”).
- Warn users before external redirects and preserve job context in-app.
- Auto-save progress and allow resume parsing + editable review.
- Use progressive disclosure for long forms and show inline examples.

2) CV/Profile Builders & Guided Input

Patterns

- Text-driven profiles (free-form summary, experience, skills).
- Structured form builders with section-based editing.
- Profile completeness indicators and missing-field prompts.

Friction Points

- Users struggle without examples and guidance on specificity.
- Too many fields shown at once increases cognitive load.
- Skill taxonomy ambiguity (e.g., Java vs JavaScript).

Recommendations

- Micro-task approach: guide users through one section at a time.
- Inline examples + smart defaults + autocomplete skill taxonomy.
- Completion + quality indicators (coverage, specificity, recency).

3) Skills Assessment & Verification

Patterns

- Self-declared skills (keyword-based).
- Optional assessments or technical screening (tech platforms).

Friction Points

- Unclear benchmarks and how scores influence matching.
- Drop-off during long assessments.
- Repetitive assessment experiences across platforms.

Recommendations

- Use short micro-assessments (2–3 minutes) with clear purpose.
- Show outcome context (percentile, confidence band, strengths).
- Let users skip with transparency: “Self-declared (unverified)”.

4) Job Matching / Recommendation Algorithms

Patterns

- Implicit matching from activity signals (views, clicks, searches).
- Explicit matching with rationale (skills, preferences, culture).

Friction Points

- Opaque recommendations reduce trust.
- Irrelevant repeat suggestions with weak feedback loops.
- Limited user controls to correct preferences.

Recommendations

- Expose match score with breakdown (skills, experience, preferences).
- Add user controls: “Not relevant because...” and learn from it.
- Allow preference tuning (location, salary, role type, stack).

5) Application Tracking Dashboards

Patterns

- Status labels (Submitted, Viewed, Interviewing, Offer).
- Simple lists or stage-based tracking.

Friction Points

- Unclear next steps and timing expectations.
- Status labels vary by employer and are often ambiguous.

Recommendations

- Use Kanban stages with consistent definitions.
- Provide next suggested action (follow-up timing, prep tasks).
- Highlight stalled applications and recommended nudges.

6) Match Score Explanations & Transparency

Patterns

- Some platforms show fit percentage, fewer explain the inputs.
- Occasional highlights of matched requirements.

Friction Points

- Scores feel arbitrary without explainability.
- Missing skills vs missing preferences is rarely separated.

Recommendations

- Split score into components: Skill Fit, Experience Fit, Preference Fit.
- Show top 3 reasons you match + top 3 gaps to address.
- Provide actionable fixes (add evidence, reword bullets, add links).

7) “Why You’re a Good Fit” Features

Patterns

- Short rationale blurbs and highlighted requirements met.
- Occasional comparison to role must-haves.

Friction Points

- Generic explanations don't build trust.
- No improvement guidance.

Recommendations

- Personalize rationale with explicit matched skills/experience.
- Show "You match X/Y must-haves" + what to add to reach Y.
- Offer one-click CV tailoring suggestions per job.

Screenshot & Annotation Checklist (Insert Your Captures)

- Job list → job detail → apply flow map (with step counts).
- Friction hotspots annotated (drop-offs, confusing fields, redirects).
- Profile builder patterns (sectioning, prompts, examples).
- Skills entry + verification / assessment screens.
- Match score UI + explanation breakdown and rationale text.
- Application tracking dashboard views (statuses, next steps).
- Feedback controls for recommendations (dismiss reasons, tuning).
- Accessibility notes (keyboard navigation, error messaging, contrast).

UI Recommendations Summary

- Reduce application friction via prefill, autosave, and progressive disclosure.
- Build trust with transparent match scoring and clear rationale.
- Offer user control to tune recommendations and correct mismatches.
- Provide actionable next steps throughout (apply, improve profile, follow up).