APROPOS – MATCH VISUALISATION REDESIGN

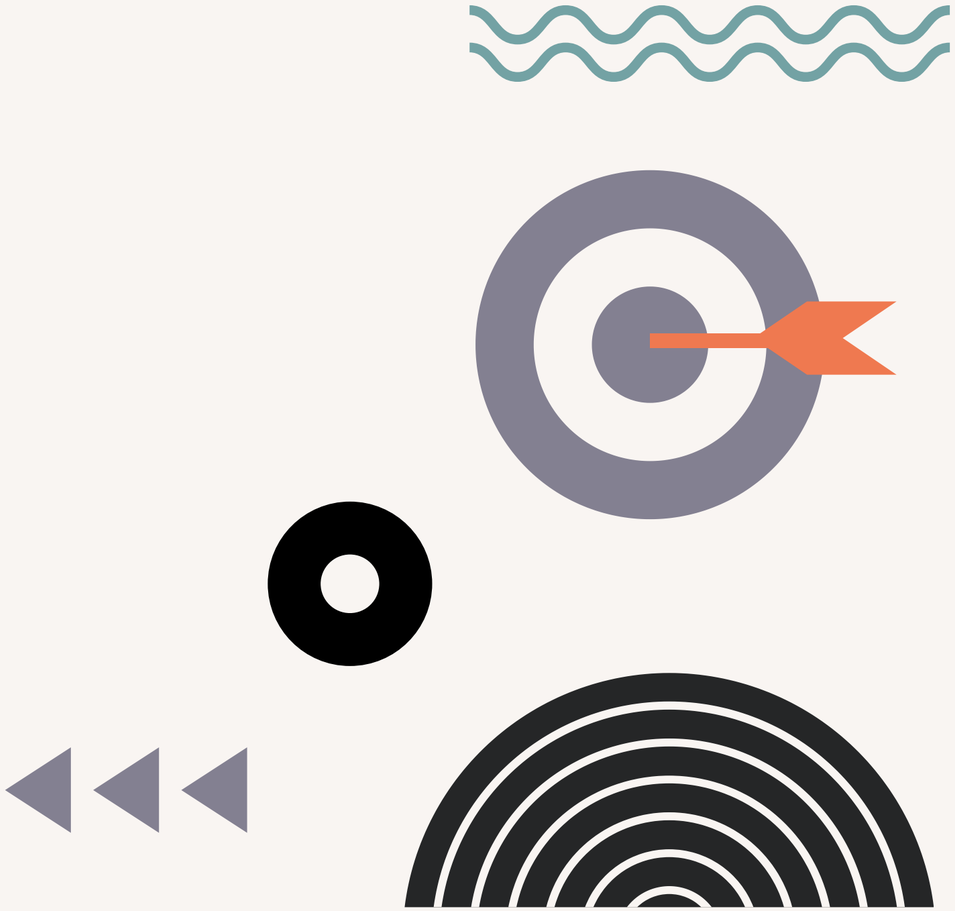
Improving clarity and confidence in personality-based job matching

prepared by Jara Belmonte

PROJECT SUMMARY

The product

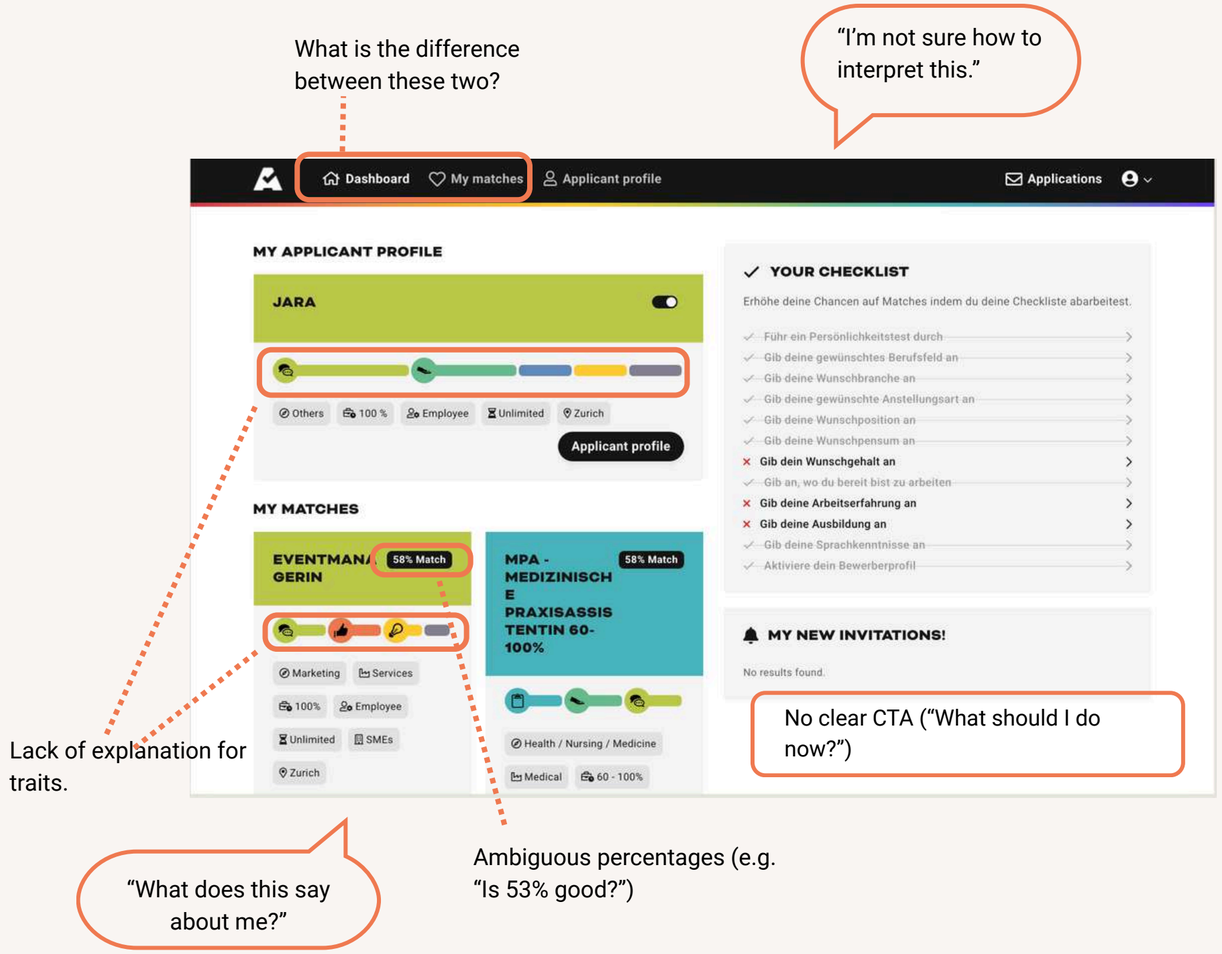
- Apropos helps align job seekers and companies using a personality-based test.
- The challenge: match results are not always clear to users.
- Goal: Make results interpretable, trustworthy, and actionable.



DEFINING THE PROBLEM

I explored the current interface and found some key friction points:

- The percentages feel flat — hard to read or interpret
- It’s unclear which traits contribute to the match
- Filters like salary or pensum are visually blended with test-based traits
- Action steps are missing — users don’t know what to do next”



PAIN POINTS

1

Pain point

Users don’t understand what a “good match” is.
> clarified scores with thresholds and natural-language summaries.

2

Pain point

No clear CTA after seeing a match.
> Introduced guidance like “Apply Now” or “Shortlist Candidate”.

3

Pain point

All profiles seem balanced — no visual hierarchy.
> Re-designed trait tables and added colour-coded insights.

4

Pain point

Although logistical filters like industry, location, workload, and salary are visually presented as separate tags (which is good), they are actually added to the final match percentage — making it unclear whether a strong match is due to personality alignment or practical filters.
> I propose keeping these dimensions visually separate and conceptually distinct from the personality score — to build transparency and trust.

>> These issues raised the need for clearer logic and stronger visual communication.

UNDERTANDING THE USER

I reviewed the platform as a user. Key assumptions included that users understood their personality test results. However, after evaluating the current UI, I noticed both job seekers and companies often didn’t understand the match percentage, nor what action to take next. This led me to restructure how the matching results are visualised and explained.

PERSONAS

Clara Abbey- the recruiter

Age:	38
Nationality:	Swiss
Location:	Zürich, Switzerland
Occupation:	HR Manager

Clara is responsible for hiring and creates job postings. She fills out the personality test to define the “ideal candidate” and reviews applicants via the dashboard.

Goals

- Find best-fit candidates
- Manage multiple job postings
- Streamline the hiring process

Needs

- Filters by job posting
- Trait comparisons
- Shortlist/Discard actions
- Quick UI per role
- Insightful match explanations

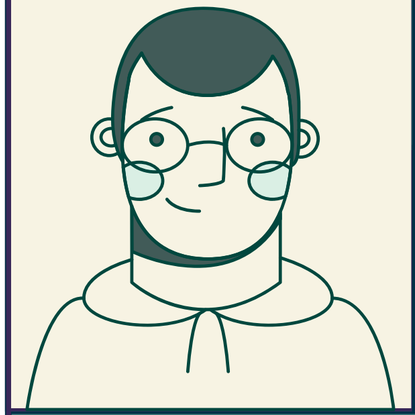
Pain Points

- Difficult to compare match %
- No overview per role
- No quick trait comparison
- Cannot annotate/respond

Technology

- Technology: Uses a computer in the office, sometimes checks data from mobile phone.

PERSONAS



David Miller - the job applicant

Age:	**42**
Nationality:	**Irish**
Location:	**Lausanne, Switzerland**
Occupation:	**Data analyst**

David is is exploring job opportunities and wants to find a company that fits his values and work style. He may:
Take the Test independently (Flow A)
Or be invited to take it by a company (Flow B)

Goals

- Discover meaningful job matches
- Understand compatibility
- Explore new roles

Needs

- Clear, digestible explanation of what the match % is based on
- Quick actions: Apply / Save / Dismiss
- Visual “why this match” summary
- Unified dashboard to track all matches
- Reassurance that this system is fair, accurate, and useful

Pain Points

- Match % unclear
- Confusing flows (self vs. company)
- Lacks clear CTAs
- No insight into why it's a match

Technology

- Technology: Uses a computer and often checks data from mobile phone.

USER JOURNEY MAP

Design adapts tone based on user flow context.

Flow A: Job Seeker - Self-initiated

Create Profile → Take Personality Test → View Matches (≥51%) → Apply / Save / Dismiss

Flow B: Job Seeker - Invited by Company

Receive Invite → Take Personality Test → See Results for That Job → Apply / Save / Dismiss

Company Flow

Create Job Offer + Target Traits → See Candidates → Filter by Role or Match % → View Profile / Save / Discard

1.Job seeker explores

Flow A: Job Seeker Takes tests on its own
Sees compatible companies
Can apply proactively

2. Flow B - Company Invitation

Flow B: Invited by Company -
Company asks you to take the test Result shown by role
You can choose: maintain interest or not

3. Flow Company - Candidate Management

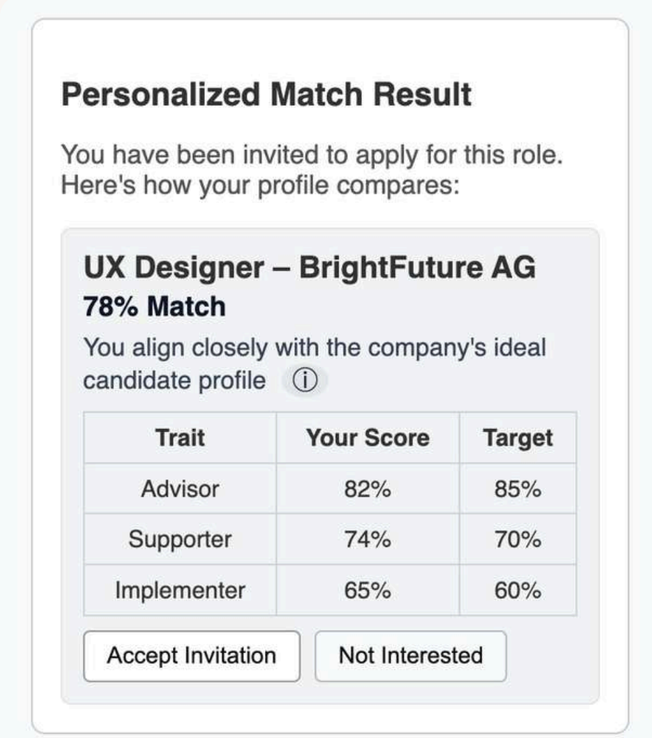
Create offer and define ideal profile -
View compatible candidates -
Can filter, discard, save, comment

FLOW A

Job Seeker A: Takes the test independently

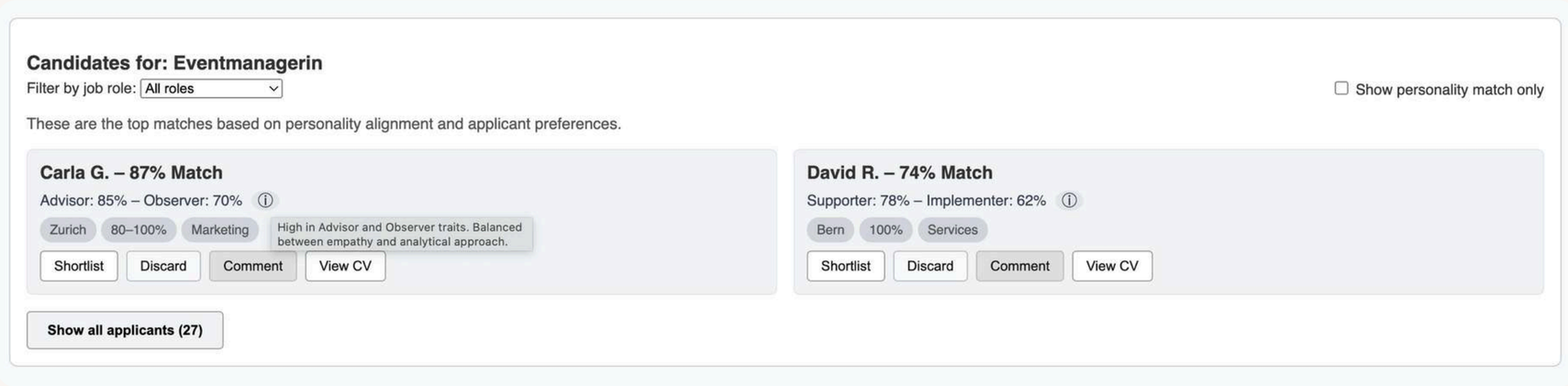
FLOW B

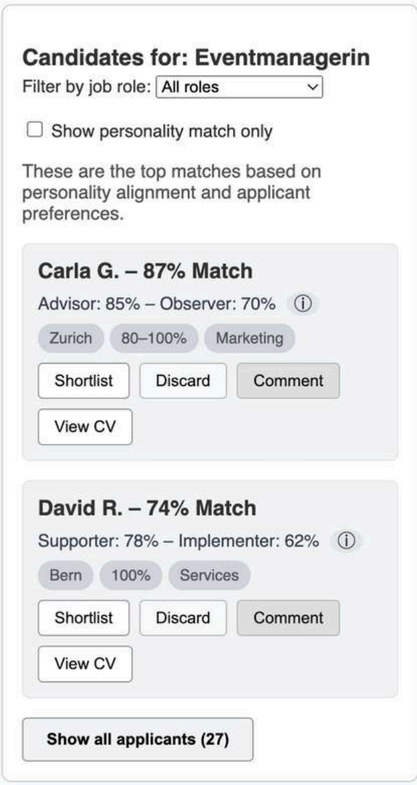
Job Seeker B: Is invited to take a test by a company



FLOW C

Company view

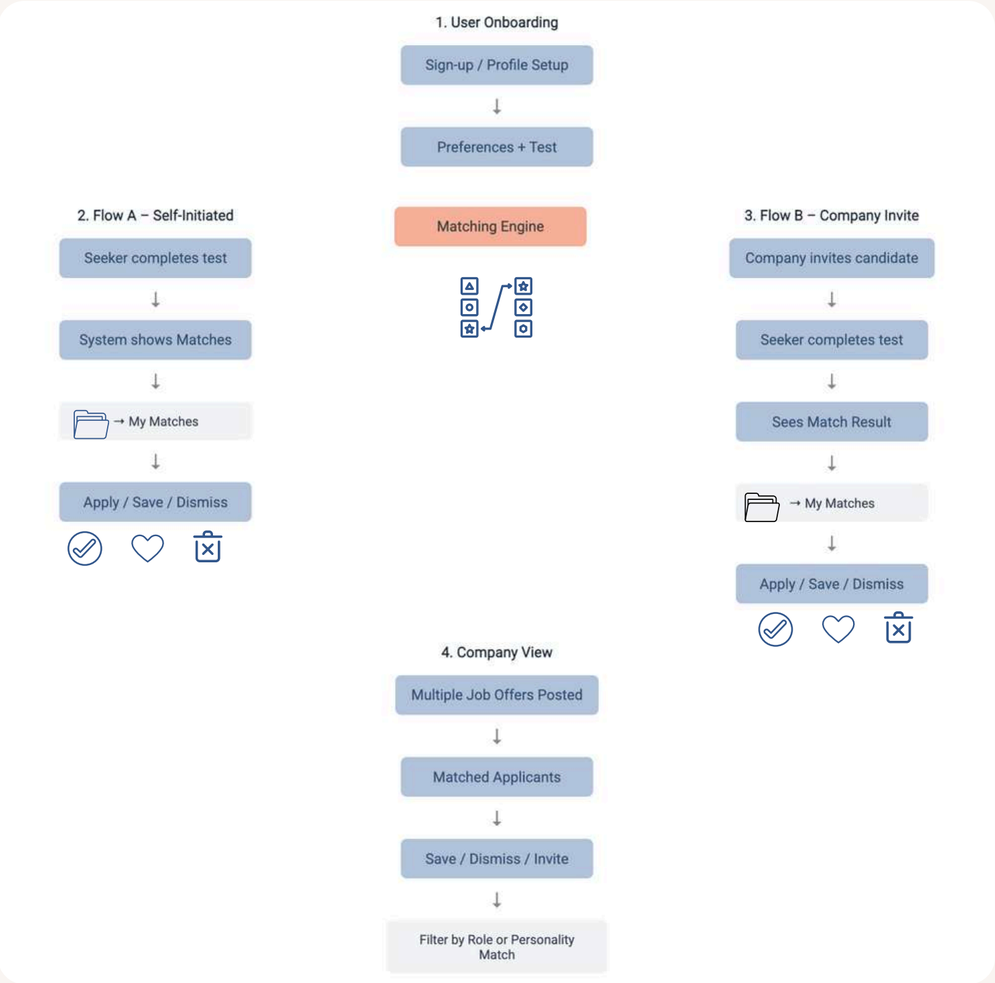




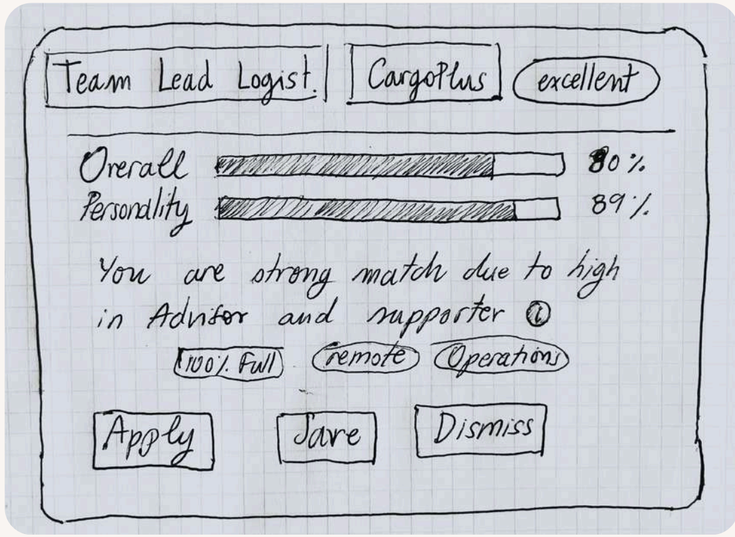
UX GLOBAL FLOW MAP

How the product works as a system.

- How test + profile data → feed into the match engine
- How results show up in “My Matches”
- How both Flow A and B converge into the same UI system
- How actions like Save / Apply / Dismiss are available
- And the company’s view that mirrors this logic from the other side



UNDERSTANDING WHAT CONTRIBUTES TO A MATCH



Key Factors

- **Personality fit**
- **Job details**
- **Skills & Experience**
- **CTA Buttons:**

Where You See Them in the UI

- **Match Bars:** Visualize global and personality alignment. Users compare overall and trait-based fit.
- **Trait Summary:** Textual insight into strongest personality traits — hover or tap for detailed tooltip.
- **Filters:** These confirm match on role-specific criteria: % pensum, location, industry, etc.
- **CTA Buttons:** Apply, Save or Dismiss jobs directly from the card. Users control next steps.

WHAT ACTIONS USERS CAN TAKE

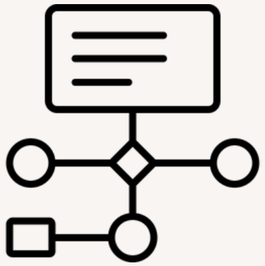
Based on Match Quality

Match Quality	Suggested Action	Visual Cue
Excellent	Apply now	"Excellent match" label + high bar
Good	Save / Apply	"Good match" label + mid-high bar
Fair	Explore, Save for later	"Fair match" label + medium bar
Low	Dismiss	"Low match" label + greyed CTA

SCALABILITY & JSON MAPPING

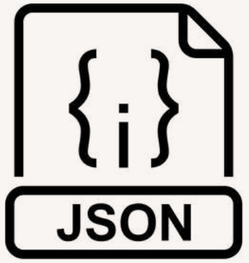
How the new component scales:

Works for both Flow A (seeker-initiated) and Flow B (company-invited)
Company view reuses same visual logic with adjusted CTAs
Mobile-responsive: Grid turns into vertical stack
Accepts real-time % values from matching engine (see JSON sample)



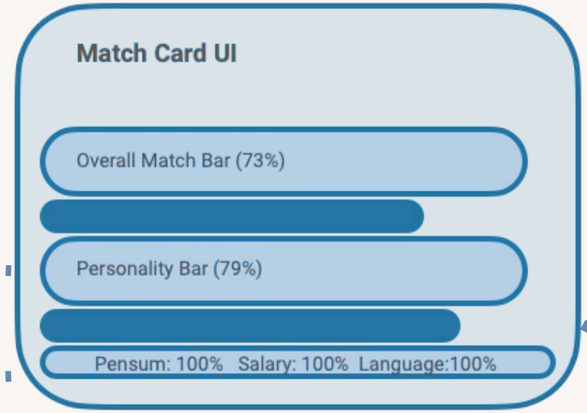
Visual mapping:

- Match % → Overall bar
- Personality % → Personality bar
- Traits → Trait summary (based on candidate data)
- Matching criteria (pensum, salary...) → Tags under card

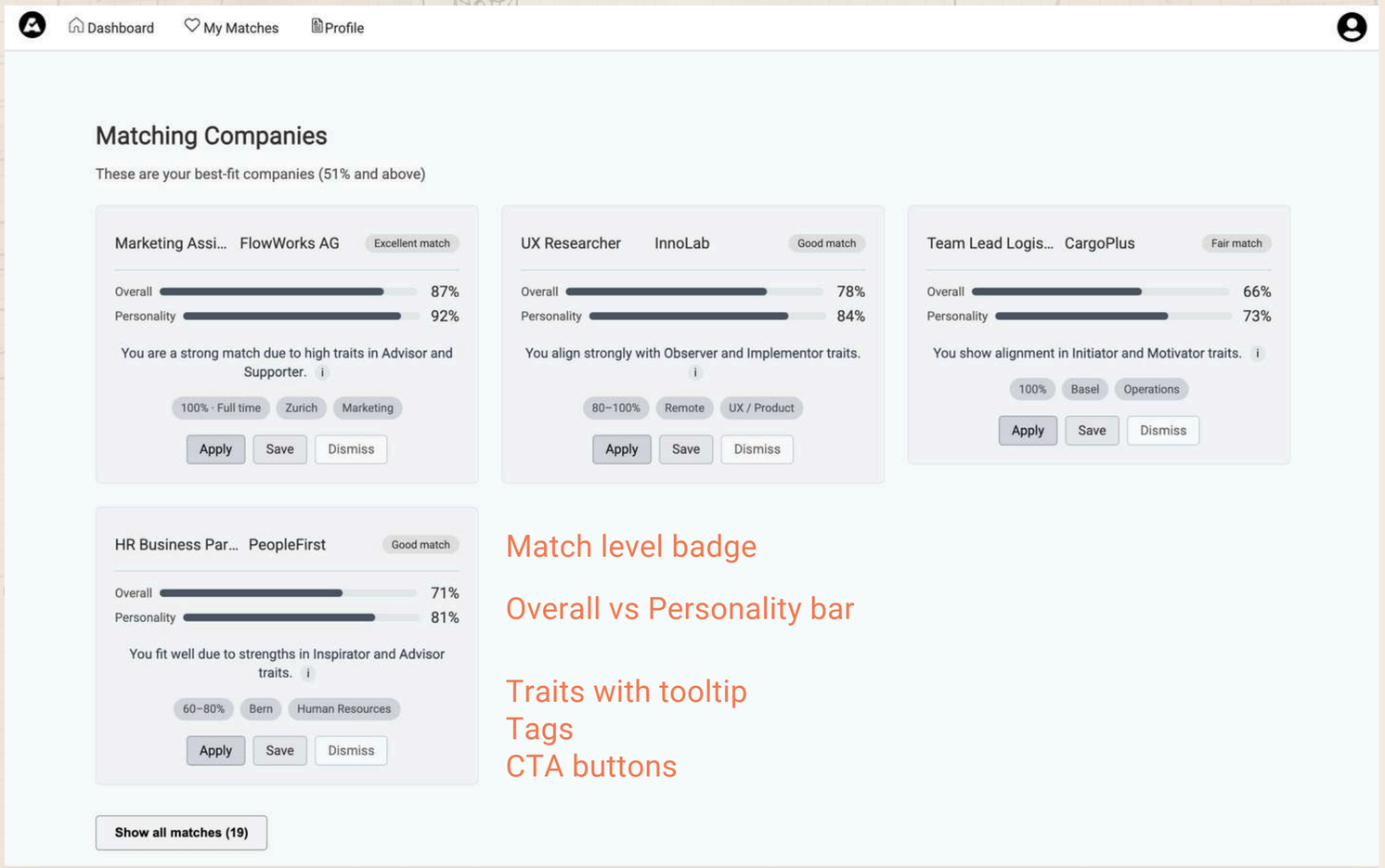


JSON compatibility Sample:

{
"match": 73,
"percents": {
"personality": 79,
"pensum": 100,
"salary": 100,
"language": 100,
"educationType": 100,
"experience": 20
}
}



LOW-FI PROTOTYPE



Match level badge

Overall vs Personality bar

Traits with tooltip
Tags
CTA buttons

[GitHub link]

[Codepen]

Initial ideation sketches used to plan layout and interactions

FINAL OUTCOMES AND NEXT STEPS

Brief overview of the case

- Builds user trust and satisfaction.
- Enhances match clarity and decision-making.
- Next: Test with real users, refine based on feedback, high-fidelity prototype.

SWIPE-BASED MATCHING

Opportunities & Design Considerations

Opportunities

Familiar UX: Engages users with a Tinder-style interface, appealing to younger demographics.

Quick Decisions: Enables rapid initial filtering, reducing cognitive load in early stages.

Anonymity: Allows users to explore opportunities discreetly.

Risks

Trust Dependency: Requires users to have complete faith in the algorithm's accuracy.

Lack of Transparency: Users may not understand why a match was made, leading to skepticism.

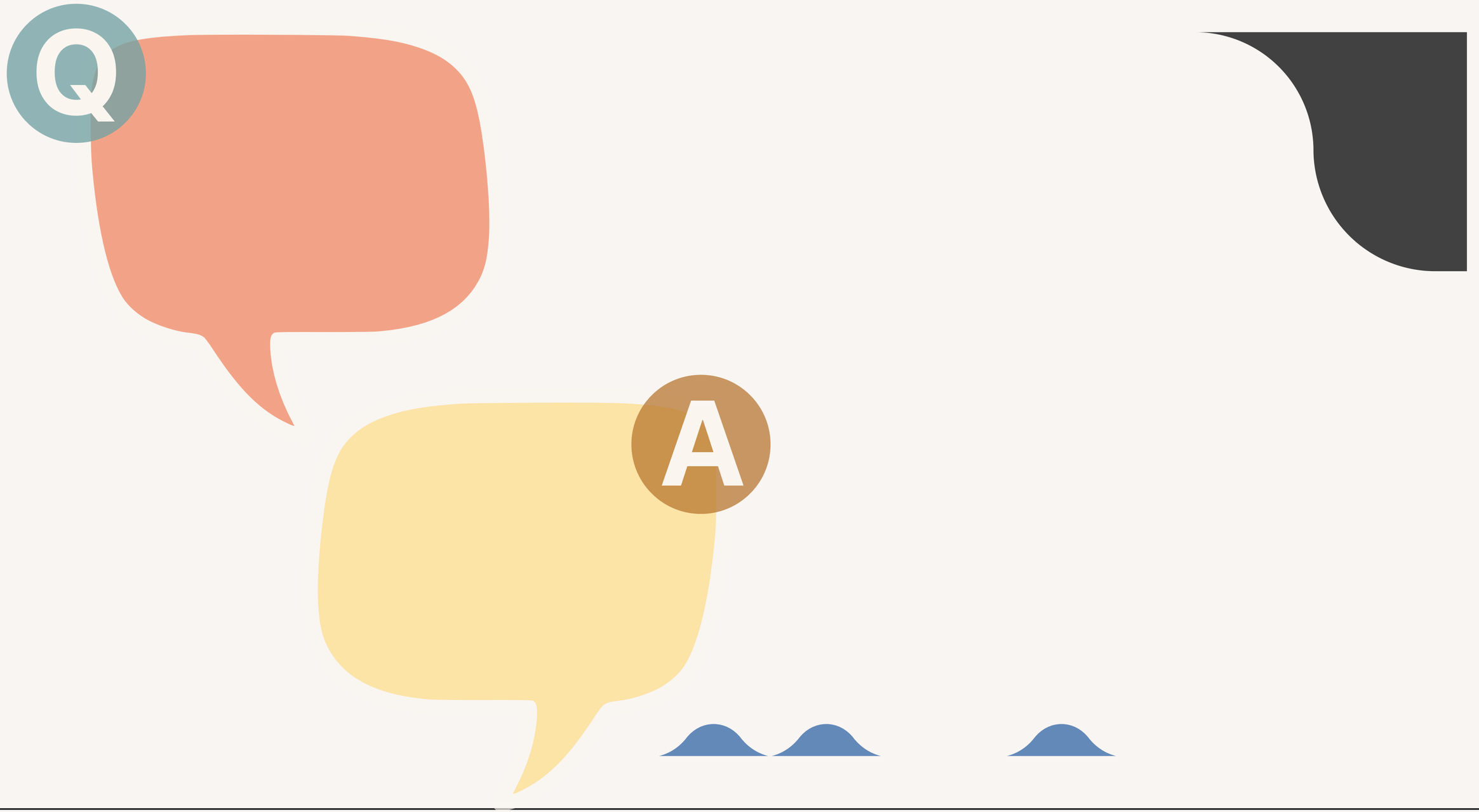
Superficial Assessment: Quick swipes may not capture the depth of personality-job fit.

Design Strategy

Explainability: Incorporate features that allow users to see why a match was made.

Hybrid Approach: Combine swipe functionality with detailed profiles for deeper insights.

User Control: Provide options to adjust matching criteria, enhancing trust and personalisation.


Q
A

THANK YOU