Survey-to-Database Alignment for Revamped Mentoring Survey

Context

The mentoring survey has been significantly revamped. The new survey introduces a **capability-based model** (30 capabilities, proficiency ratings, practice scenarios) replacing the old **topic-category model** (~10 broad topics, life experiences). This document maps every new survey question to database fields, identifies gaps, and outlines the work to align the import pipeline.

---

1. Full Survey → Database Field Mapping

Shared Bio Fields (Q4–Q8, both roles)

#	Survey Question	Current DB Field	Status	Action
Q4	Role title	`role`	☐ Exists	Update parser header matching
Q5	Function (~60 dept options)	`industry` (loosely)	☐☐ Mismatch	Repurpose `industry` → store function, or add `function_area` column
Q6	Preferred timezone	`location_timezone`	☐ Exists	Update parser for new option labels
Q7	Current level (S1–LT)	`seniority_band` (computed)	☐☐ Indirect	Store directly in `seniority_band` instead of computing from `experience_years`
Q8	Bio (5–8 lines)	*(none)*	☐ Missing	**Add `bio` text column** to both tables
Q9	Role selection (Mentee/Mentor/Both)	Auto-detected	☐☐ Implicit	Parser should use this column explicitly

Mentee Section (Q10–Q22)

#	Survey Question	Current DB Field	Status	Action
Q10	Primary capability (1 of 30)	`topics_to_learn[0]`	☐☐ Structural change	**Add `primary_capability` text column**
Q11	Domain expertise detail (conditional)	*(none)*	☐ Missing	**Add `primary_capability_detail` text column**
Q12	Secondary capability (optional)	`topics_to_learn[1]`	☐☐ Structural change	**Add `secondary_capability` text column**
Q13	Domain expertise	*(none)*	☐ Missing	**Add secondary_capability_detail**

#	Survey Question	Current DB Field	Status	Action
Q13	detail (conditional)	*(none)*	☐ Missing	`secondary_capability_detail` **text column**
Q14	Primary proficiency (1–4)	*(none)*	☐ Missing	**Add** `primary_proficiency` **integer column**
Q15	Secondary proficiency (1–4)	*(none)*	☐ Missing	**Add** `secondary_proficiency` **integer column**
Q16	Mentoring goal (structured)	`motivation` / `expectations`	☐☐ Different concept	**Add** `mentoring_goal` **text column**
Q17	Practice scenarios (pick 2 of 12)	*(none)*	☐ Missing	**Add** `practice_scenarios` **text[] column**
Q18	Real situation for session 1	*(none)*	☐ Missing	**Add** `real_situation` **text column**
Q19	Mentor help wanted (pick 2 of 7)	`mentor_qualities`	☐☐ Similar	Repurpose `mentor_qualities` or **add** `mentor_help_wanted` **text[]**
Q20	Open to first-time mentor	`mentor_experience_importance`	☐ Same concept	Repurpose — update parser header
Q21	Session style preference	`preferred_style`	☐ Same concept	Repurpose — update parser header
Q22	Feedback style preference	`feedback_preference`	☐ Same concept	Repurpose — update parser header

Mentor Section (Q23–Q41)

#	Survey Question	Current DB Field	Status	Action
Q23	Open to multiple mentees (Yes/No/Depends)	`capacity_remaining` (number)	☐☐ Different format	**Add** `open_to_multiple` **text column**
Q24	How many (conditional)	`capacity_remaining`	☐ Exists	Update parser
Q25	First time mentoring	`has_mentored_before` (boolean)	☐☐ Richer answer	**Add** `mentoring_experience` **text column** (keep boolean for compat)
Q26	First-time support needed (pick 2)	*(none)*	☐ Missing	**Add** `first_time_support` **text[] column**
Q27	Primary capability (1 of 30)	`topics_to_mentor[0]`	☐☐ Structural change	**Add** `primary_capability` **text column**
Q28	Domain expertise detail (conditional)	*(none)*	☐ Missing	**Add** `primary_capability_detail` **text column**
Q29	Secondary capability (optional)	`topics_to_mentor[1]`	☐☐ Structural change	**Add** `secondary_capability` **text column**
Q30	Domain expertise detail (conditional)	*(none)*	☐ Missing	**Add** `secondary_capability_detail` **text column**
Q31	Primary proficiency (1–5)	*(none)*	☐ Missing	**Add** `primary_proficiency` **integer column**
Q32	Secondary proficiency (1–5)	*(none)*	☐ Missing	**Add** `secondary_proficiency` **integer column**

Q33	Practice scenarios (pick up to 4)	*(none)*	☐ Missing	**Add `practice_scenarios` text[] column**
Q34	Real example of handling scenario	*(none)*	☐ Missing	**Add `mentor_example` text column**
Q35	Hard-earned lesson	*(none)*	☐ Missing	**Add `hard_earned_lesson` text column**
Q36	Natural strengths (pick 3 of 7)	`mentoring_style`	☐☐ Different concept	**Add `natural_strengths` text[] column**
Q37	Session style	`meeting_style`	☐ Same concept	Repurpose — update parser header
Q38	Capabilities NOT to mentor (multi-select)	`topics_not_to_mentor`	☐ Same concept	Repurpose — parser update
Q39	Feedback style	`feedback_style`	☐ Same concept	Repurpose — update parser header
Q40	Scenarios NOT to support (multi-select)	*(none)*	☐ Missing	**Add `excluded_scenarios` text[] column**
Q41	Match exclusions (open text)	*(none)*	☐ Missing	**Add `match_exclusions` text column**

---

2. Summary of New DB Columns Needed

Mentees table — 10 new columns

Column	Type	Nullable
`bio`	text	yes
`primary_capability`	text	yes
`primary_capability_detail`	text	yes
`secondary_capability`	text	yes
`secondary_capability_detail`	text	yes
`primary_proficiency`	integer	yes
`secondary_proficiency`	integer	yes
`mentoring_goal`	text	yes
`practice_scenarios`	text[]	yes
`real_situation`	text	yes

Mentors table — 16 new columns

Column	Type	Nullable
`bio`	text	yes
`open_to_multiple`	text	yes
`mentoring_experience`	text	yes
`first_time_support`	text[]	yes
`primary_capability`	text	yes
`primary_capability_detail`	text	yes
`secondary_capability`	text	yes
`secondary_capability_detail`	text	yes
`primary_proficiency`	integer	yes

`secondary_proficiency`	integer	yes
`practice_scenarios`	text[]	yes
`mentor_example`	text	yes
`hard_earned_lesson`	text	yes
`natural_strengths`	text[]	yes
`excluded_scenarios`	text[]	yes
`match_exclusions`	text	yes

---

3. Fields Becoming Obsolete

These DB fields exist but the new survey no longer collects them:

Field	Table(s)	Reason
`life_experiences` + boolean flags	Both	Life experience questions fully removed
`experience_years`	Both	Replaced by direct level (Q7 → `seniority_band`)
`languages`	Both	Not asked in new survey
`meeting_frequency`	Both	Program is now fixed schedule
`preferred_energy` / `mentor_energy`	Mentee / Mentor	Not asked
`main_reason`	Mentee	Absorbed into structured `mentoring_goal`
`unwanted_qualities`	Mentee	Not asked
`other_experience(s)`	Both	Not asked

Recommendation: Keep old columns (nullable) for backward compat with existing cohorts. New cohorts will leave them null.

---

4. Matching Algorithm — New Design

New scoring weights

Component	Weight	Fields	Notes
Topics Match	45%	`primary_capability`, `secondary_capability`, `practice_scenarios`	Primary-to-primary = full score, cross-match (primary/secondary) = partial. Practice scenario overlap as bonus within this bucket.
Semantic Similarity	30%	Mentee: `mentoring_goal` + `real_situation` + `bio`. Mentor: `mentor_example` + `hard_earned_lesson` + `bio`	Story + domain detail + goal text is powerful for tie-breaking and nuance
Domain Match	5%	`primary_capability_detail`, `secondary_capability_detail`	Text similarity on the domain expertise free-text fields. Rewards specificity overlap (e.g. both mention "revenue ops").
Seniority Fit	10%	`seniority_band` (S1–LT, now directly from survey Q7)	Mentor should be >=1 level above mentee. Same-level or below = penalty.
Timezone Bonus	5%	`location` `timezone`	Same TZ = full bonus, <=2h diff

—

			= partial, >2h = zero
Capacity Penalty	-10%	`capacity_remaining`	Applied when mentor has only 1 slot left, encouraging spread

Removed from old algorithm

- ~~Language (5% + hard filter)~~ — removed entirely, everyone speaks English
- ~~Industry (15%)~~ — removed, everyone is at Mews

Hard filters (exclude before scoring)

Filter	Logic	Source
Commitment check	If either person answered No to any of Q1–Q3 → exclude from matching	Form blocks them anyway, but safety net
Domain expertise detail required	If mentee picked "Domain Expertise" as their capability but left the detail field empty → **flag for review** + lower score (don't hard-exclude, but penalise)	`primary_capability` = "Domain Expertise" AND `primary_capability_detail` is empty
Geographic hard block	If timezone difference >6 hours → **don't match**	`location_timezone` — numeric hour-based threshold (replaces old 3h soft limit)
Capacity exhausted	Mentor at max capacity (`capacity_remaining` = 0) → **exclude**	Already exists, keep
Excluded scenarios	If mentee's `practice_scenarios` overlap with mentor's `excluded_scenarios` → **don't match**	New hard filter
Match exclusions	If mentor's `match_exclusions` free-text flags something relevant → **flag for manual review**	Admin review, not auto-scored

---

5. Other Affected Areas

Profile Display (ManualMatchingBoard, MenteeProfile, MentorProfile)

- Update to show new fields (bio, capabilities, proficiency, practice scenarios, goal)
- Remove or hide obsolete fields for new cohorts
- Life experience section no longer relevant for new data

Analytics (PeopleAnalytics)

- Topic demand/supply charts → Capability demand/supply
- Life experience distribution chart → Remove or skip for new cohorts
- New chart opportunities: proficiency distribution, practice scenario coverage

CSV Parser (dataParser.ts)

- Clean break: replace old parser with new survey column mappings

- New column header matching for MS Forms full question text headers
- Handle multi-select answers (MS Forms semicolons, or boolean columns)
- Handle "Both" respondents (one row → split into mentee + mentor records)

TypeScript Types

- Update `MenteeData` and `MentorData` interfaces in `src/types/mentoring.ts`
- Update `Database` types in `src/types/database.ts`

Power Automate Webhook (new — mirrors `log-session` pattern)

- New Supabase Edge Function: `import-survey-response`
- Receives individual form submissions in real-time as people complete the survey
- Same dual-path approach as session logs: automated webhook OR manual CSV bulk import
- See Section 6, Phase F for details

---

6. Implementation Phases

Phase A: Database & Types

1. Write Supabase migration adding all new columns to both tables
2. Update `src/types/database.ts` with new columns
3. Update `src/types/mentoring.ts` with new interface fields

Phase B: CSV Parser (clean break — replace old parser functions)

4. Replace `parseMenteeRow()` with new column mappings for the revamped survey
5. Replace `parseMentorRow()` with new column mappings
6. Use Q9 "Role selection" column for explicit mentor/mentee detection
7. Auto-detect multi-select format (semicolon-separated vs boolean columns)
8. Also populate legacy array fields (`topics_to_learn`/`topics_to_mentor`) from primary+secondary capabilities so matching/analytics still have array data

Phase C: Database Service

9. Update `supabaseService.ts addImportDataToCohort()` to persist all new fields

Phase D: Matching Algorithm (see Section 4 for full design)

10. Replace scoring weights: Topics 45%, Semantic 30%, Domain 5%, Seniority 10%, TZ 5%, Capacity -10%
11. Remove language filter + language bonus + industry scoring entirely
12. Implement capability matching (primary-to-primary full, cross-match partial, practice scenario overlap)
13. Implement domain detail text similarity (5% bucket) on `primary_capability_detail`
14. Update `embeddingUtils.ts` — mentee: `mentoring_goal` + `real_situation` + `bio`; mentor: `mentor_example` + `hard_earned_lesson` + `bio`
15. Add hard filters: >6h timezone block, excluded scenarios, domain detail flag, commitment check
16. Update `seniority_band` scoring to use direct S1–LT levels from survey

Phase E: UI Updates

17. Update ManualMatchingBoard profile previews
18. Update MenteeProfile / MentorProfile components
19. Update PeopleAnalytics charts

Phase F: Power Automate Webhook (mirrors `log-session` pattern)

Create `supabase/functions/import-survey-response/index.ts` following the existing `log-session` edge function pattern:

20. **Endpoint**: `POST /functions/v1/import-survey-response`
21. **Auth**: API key header (`SURVEY_IMPORT_API_KEY` in Supabase secrets)
22. **Payload**: Single form response with all survey fields mapped to snake_case keys. Power Automate sends one POST per form submission.
23. **Logic**:
    o Validate API key
    o Determine role from Q9 (`mentee`, `mentor`, or `both`)
    o Resolve target cohort (use the active/draft cohort, or accept `cohort_id` param)
    o Map form fields → DB columns (same mapping as CSV parser, shared mapping function)
    o If `both`: upsert into mentees table AND mentors table
    o Deduplicate: upsert on `(cohort_id, mentee_id)` / `(cohort_id, mentor_id)` so re-submissions update rather than duplicate
24. **Shared mapping**: Extract field-mapping logic into a shared module used by BOTH the CSV parser and the edge function, so mappings stay in sync
25. **Response**: Return success with created/updated record IDs, or error with details

Power Automate flow setup (external to codebase, but documented):

- Trigger: "When a new response is submitted" (MS Forms)
- Action: "Get response details"
- Action: "HTTP POST" to edge function URL with API key header
- Map each form question to the expected JSON field name

Phase G: Cohort Message Hub (integrated into Runbook)

Recommendation: Build into the existing Runbook, not a separate section.

The Runbook already has 8 lifecycle stages (Setup → Import → Matching → Review → Launch → Midpoint → Closure → Reporting) and checklist items that mention communications ("Send match notifications", "Distribute welcome materials"). Right now those are just manual reminders. This phase turns them into actual message generation.

Architecture: Runbook-Integrated Message System

How it works:

- Each Runbook stage can have **message templates** attached to it
- When admin reaches a stage, they click "Generate Messages" to produce personalised, copy-pasteable messages for all relevant participants
- Messages are populated from participant profile data (the app already has everything needed)
- Admin copies messages into Slack (since the app is not IT-sanctioned, no direct Slack API)
- Message generation history is logged so you can see what was sent when

Messages by Cohort Stage

Stage	Message	Recipients	Key Personalisation Fields
Launch	Pair intro message	Each matched pair	Mentee goals, mentor strengths, capabilities, style
Launch	Welcome + resources	All participants	First name, role (mentor/mentee), resource links
Post-Session 1	"How was your first session?" nudge	All pairs (triggered ~2 weeks after launch)	First name, partner name
Midpoint	Mid-cohort check-in	All participants	First name, session count so far, upcoming milestones
Midpoint	Session log reminder	Pairs with no recent logged session	First name, last session date, partner name
Closure	Final survey + thank you	All participants	First name, partner name, session count, capability worked on
Closure	Re-match offer (if needed)	Pairs flagged as not working	First name, admin contact
Reporting	Program outcomes summary	All participants	Cohort stats, personal stats

Message Template System

30.

Template storage: New `message_templates` table or JSONB field on `cohort_stages`

- o Each template has: `stage`, `name`, `audience` (all/mentees/mentors/pairs), `body` with placeholders
- o Default templates ship with the app (editable per cohort)
- o Admin can customise templates before generating

31.

Available placeholders (populated from profile data already in the app):

Person fields:

- o `{FIRST_NAME}`, `{FULL_NAME}`, `{ROLE_TITLE}`, `{FUNCTION}`
- o `{SLACK_HANDLE}` — derived from email or new DB field
- o `{PRIMARY_CAPABILITY}`, `{SECONDARY_CAPABILITY}`
- o `{MENTORING_GOAL}` (mentee), `{BIO}`
- o `{SESSION_STYLE}`, `{FEEDBACK_STYLE}`

Mentor-specific:

- o `{NATURAL_STRENGTHS}`, `{HARD_EARNED_LESSON}`
- o `{MENTOR_EXAMPLE}`, `{MENTORING_EXPERIENCE}`

Pair fields:

- o `{MENTEE_FIRST_NAME}`, `{MENTOR_FIRST_NAME}`
- o `{MENTEE_SLACK_HANDLE}`, `{MENTOR_SLACK_HANDLE}`
- o `{SHARED_CAPABILITY}` — the capability they matched on
- o `{SESSION_COUNT}` — sessions logged so far

Cohort fields:

- o `{COHORT_NAME}`, `{COHORT_START_DATE}`, `{COHORT_END_DATE}`
- o `{ADMIN_SLACK_HANDLE}` — configurable per cohort
- o `{RESOURCE_LINKS}` — configurable list of links per cohort

32.

UI — integrated into Runbook page:

- o Each stage shows a "Messages" tab alongside Checklist and Documents
- o "Generate Messages" button produces all messages for that stage
- o Messages shown in a scrollable list, grouped by pair or person
- o Each message has **"Copy to clipboard"** button
- o **"Copy All"** button for batch pasting
- o **"Mark as Sent"** button to track what's been sent
- o Edit button to tweak individual messages before copying

33.

Communication log:

- o Track which messages were generated and marked as sent
- o Visible per cohort: "Launch intro messages — sent 15 Feb 2026 — 12 pairs"
- o Helps the admin team know what's been done vs outstanding

34.

Smart personalisation for pair intros (the richest message):

- o Pull mentee's `mentoring_goal` or `bio` for the personal note
- o Pull mentor's `natural_strengths` and capabilities for their description
- o Mention the specific capability match that connected them
- o Include practice scenarios if relevant

Example pair intro message (auto-generated):

Hello! I'm really happy to introduce you as mentee @alice.smith and mentor @bob.jones. Thank you both for leaning into this adventure - I'm excited to see what you build together.

Alice is looking to work on Strategic Communication (and also Stakeholder Management). One thing that stood out is her goal to improve how she presents to senior leadership in cross-functional settings.

Bob brings a coaching-first approach and experience in Strategic Communication / Influencing Without Authority, which feels like a really strong match for what Alice wants to explore.

I'll be adding you shortly to two separate Slack channels — we'll share a few resources and quick surveys along the way to keep improving the program.

Mentoring resources:

- Mentoring guide: [link]
- Session log form: [link]

First step: @alice.smith — when you can, please set up a first call with @bob.jones.

For the first session, keep it simple and human:

- Quick intros + how you like to communicate
- Alice shares goals, context, and what "success" would look like
- Bob listens, asks questions, and helps sharpen what to focus on
- Together, align on cadence, timing, communication preferences, and any boundaries

Most importantly: use that first chat to check if this feels like a fit. If either of you feels it's not quite the right match, no stress — please contact @irene.admin and we'll sort it out.

35. **Slack handle**: Either add a `slack_handle` field to the survey/DB, or derive from email (e.g. `alice.smith@mews.com → @alice.smith`). Need to confirm Mews Slack naming convention.

---

7. Decisions Made

- **Clean break** — Only new survey format supported going forward. Old cohorts keep data but no need for dual-format parser.
- **Language filter** — Remove entirely from matching (everyone at Mews speaks English).
- **Capability/scenario lists** — Will be provided later. Parser will be flexible (match by column header text, no hardcoded enum validation).
- **CSV multi-select format** — MS Forms typically uses semicolons. Parser will handle both semicolon-separated single columns AND separate boolean columns (auto-detect).
- **"Both" respondents** — One row per person with all columns. If someone selects both roles, their mentee columns (Q10–22) and mentor columns (Q23–41) are both populated in the same row. Parser must split this into one mentee record + one mentor record.
- **Survey tool** — Microsoft Forms. Column headers will be the full question text.
- **Timezone hard block** — Any pair with >6 hour timezone difference is excluded from matching.
- **Dual ingestion** — Both Power Automate webhook (real-time) and CSV bulk import (manual) supported, mirroring the existing session log pattern.

---

8. Data Privacy Assessment

Good news: Major GDPR risk removed

The **old** survey collected special category personal data (health challenges, menopause, parental leave, career breaks). The **new** survey has **removed all life experience questions** — this eliminates the most legally sensitive data category.

Sensitive free-text fields in the new survey

These fields could contain names of colleagues, internal conflicts, career frustrations, or confidential business situations:

Field	Risk	Recommendation
Bio (Q8)	Medium — "real story behind where you are" could include personal struggles	Admin-only visibility. Don't surface in analytics.
Real situation (Q18)	**High** — names colleagues, describes conflicts, "where you get stuck"	Admin + matched mentor only. Consider not persisting long-term.

Mentoring goal (Q16)	Low-Medium — structured format limits sensitivity	Fine for profile display to matched partner.
Mentor example (Q34)	Medium — "situation, stakes" could reference confidential business	Admin + matched mentees only.
Hard-earned lesson (Q35)	Low — general wisdom, unlikely to be sensitive	Fine for profile display.
Match exclusions (Q41)	Medium — "areas of business, seniority" could imply interpersonal issues	Admin-only. Never surface to other participants.

MS Forms metadata

Microsoft Forms automatically adds **Timestamp** and **Email** columns. The email is PII but necessary for participant identification. The parser already handles email — just ensure it's stored in `email` field and not exposed in analytics.

Recommendations

1. **No regulatory blockers** — nothing in the new survey is special category data under GDPR. Standard personal data protections apply.
2. **Access controls** — Free-text fields (bio, real situation, mentor example, match exclusions) should only be visible to admins and the matched partner, not to all participants or in analytics/exports.
3. **Data retention** — Consider whether `real_situation` (Q18) needs to persist after the cohort ends, since it describes a specific point-in-time workplace conflict.
4. **No new "serious trouble" risks** — The biggest risk with the old survey (health/life experience data) has been eliminated. The remaining data is standard HR-adjacent information.

---

9. Verification Plan

- **CSV import**: Import a sample CSV with the new survey structure → confirm all fields parsed and stored
- **Webhook**: POST a sample JSON payload to the edge function → confirm record created/updated
- **Re-submission**: POST the same respondent again → confirm upsert (update, not duplicate)
- **"Both" role**: Test a respondent who selected both mentor + mentee → confirm records in both tables
- **Old cohort data**: Verify existing cohorts still load and display correctly
- **Matching**: Run matching on a new cohort → verify capability matching, hard filters, new weights
- **Profile views**: Check new fields display correctly in ManualMatchingBoard and profile components
- **Analytics**: Check charts reflect new data structure
- **Message generation**: On each Runbook stage, generate messages → verify all placeholders filled with correct data → copy to clipboard works
- **Pair intro**: Finalise matches → generate intro messages → check personalisation pulls the right fields
- **Communication log**: Generate + mark as sent → verify log entry recorded with timestamp

---

Key Files to Modify

- `database/` — New migration SQL file
- `src/types/database.ts` — DB type definitions
- `src/types/mentoring.ts` — App type definitions
- `src/lib/dataParser.ts` — CSV parsing logic (~1,271 lines)
-

- `src/lib/supabaseService.ts` — DB insertion (~1,550 lines)
- `src/lib/surveyFieldMapping.ts` — **NEW** shared mapping module (used by CSV parser + edge function)
- `src/lib/matchingEngine.ts` — Matching algorithm
- `src/lib/embeddingUtils.ts` — Embedding text builder
- `supabase/functions/import-survey-response/index.ts` — **NEW** edge function (mirrors `log-session` pattern)
- `src/components/ManualMatchingBoard.tsx` — Profile previews in matching UI
- `src/components/MenteeProfile.tsx` / `MentorProfile.tsx` — Profile display
- `src/pages/admin/PeopleAnalytics.tsx` — Analytics charts
- `src/lib/messageGenerator.ts` — **NEW** template engine with placeholder substitution for all cohort messages
- `src/lib/messageTemplates.ts` — **NEW** default message templates per runbook stage
- `src/components/admin/RunbookMessages.tsx` — **NEW** message generation UI (integrated into Runbook stage view)
- `src/pages/admin/CohortRunbook.tsx` — Update to add "Messages" tab per stage