Handz V1 PRD Bundle

CH34 — Data Export & Account Deletion

Data export and account deletion requirements for App Store compliance and user trust. This chapter defines what data exists, what is exportable, how exports are generated and delivered, how deletions work end-to-end, and what copy and error states must appear so there is zero guesswork during implementation.

Required Chapter Header (per CH00)

- **Doc ID:** HZ-V1-CH34_Data_Export_Account_Deletion_R1
- **Revision:** R1 (2026-01-02)
- **Status:** Draft
- **Depends on:** CH07 (Authentication & Account System), CH08 (Entitlements & Plan States), CH28 (Offline Behavior & Sync), CH29 (Data Storage & Limits), CH31 (Error States)
- **Related:** CH15 (Library), CH16 (Flow Detail), CH17-CH19 (Sharing/Inbox/Import Conflicts), CH20-CH24 (Practice/Mastery/Maintenance), CH25 (Monetization), CH30 (Safety/Abuse Limits), CH33 (Analytics & Metrics)
- **Supersedes:** None (first release)
- **Owned Decisions:** Export package contents and formats; export UX; deletion UX; deletion semantics and retention; re-auth requirements; rate limits for exports/deletions.
- **Open Questions / Placeholders:** See §12 (Placeholder Registry).

This chapter is authored under the bundle rules in CH00 (Master Index & Manifest). Cross-references are mandatory; do not redefine rules owned by other chapters.

1. Purpose and Scope

Handz allows strikers and coaches to create moves, build flowcharts (flows), drill selected paths in Practice Mode, and track progress. This chapter specifies the **user-facing controls** and **system behavior** for two high-stakes areas:

- **Data Export:** A user can obtain a portable copy of their data (flows, moves, practice logs, mastery/gameplans, etc.) in structured formats. Exports must be understandable, complete, and secure.
- **Account Deletion:** A user can delete their account in-app. Deletion must be explicit, safe against accidents, and complete (including media) while honoring minimal retention rules.

Out of scope for CH34: Pricing, trial behavior, paywall placement, and entitlement rules are owned by CH25/CH08. Export and deletion must respect those rules but do not define them.

1.1 Non-negotiable outcomes

- A user can initiate **account deletion** from within the iOS app without contacting support.
- A user can initiate **data export** from within the app and receive a usable package.
- No data export or deletion flow introduces new security holes (account takeover amplification, mass scraping, etc.).
- Exports and deletions produce deterministic, testable outputs (acceptance tests in §13).

1.2 Definitions

- **Export Package:** A .zip file containing structured data files (JSON + CSV) and an index/manifest describing the contents and schema version.
- **Personal Data:** Data tied to a user account (profile, settings, content authored by the user, practice history).
- **Derived Data:** App-generated summaries derived from personal data (counts, streaks, mastery status).
- **Aggregated Analytics:** De-identified aggregate metrics owned by CH33 (may be retained after deletion).
- **Hard Delete:** Data is physically removed (or irreversibly anonymized) from primary stores.
- **Soft Delete:** Data is hidden and scheduled for purge after a grace period; user may undo within that window (only if we choose to support undo).

2. Data Inventory (What exists in Handz V1)

This section enumerates all data types that may exist for a user and whether each must be exportable and/or deletable. Exact field schemas are owned by feature chapters (moves, flows, practice, mastery). CH34 defines the **export surface** and **deletion surface**.

2.1 Required export categories

An export is considered complete if it includes, at minimum, all items below (if present for the user).

- **Account:** account identifiers, sign-in methods present (Apple/Google/Email), account creation date, last login date (if stored), plan state at export time (Guest/Free/Trial/Pro).
- **Profile:** display name, username/handle, any user-selected training styles/filters chosen during onboarding.
- **Settings:** app settings (notifications toggles, default filters, UI preferences) and any privacy toggles.
- **Moves library:** default-move selections (what the user enabled), user-created custom moves, user customizations to defaults (notes, tags, videos/links, aliases).
- **Flows:** all saved flows (within plan limits), folders/organization, flow metadata, and flow graphs (nodes/edges) including optional sequences and per-edge detail.
- **Sharing:** unlisted share links created by the user (active + revoked if tracked), and link metadata (created_at, revoked_at).
- **Inbox:** received imports (inbox items), their statuses (viewed, saved to library, expired).
- **Practice:** practice sessions history, per-session configuration (selected paths, order), completion state (completed/interrupted), and summary metrics.
- **Mastery/Gameplans:** gameplans created, selected paths/flows, mastery status per path (as stored), user adjustments (manual downgrade or reset).
- **Maintenance:** maintenance schedules/plans, user-defined load preferences, reminder settings, maintenance history if tracked.
- **Notes:** any freeform notes tied to moves, sequences, paths, flows, or gameplans.

2.2 Media data types

Handz V1 supports (a) shareable links and (b) Pro-only uploaded videos stored privately and not shared (CH29 lock). Export must handle both safely.

- **Link-based media (shareable):** YouTube links, private/unlisted links, or external URLs stored as references. Export includes the URLs and the attachment context.
- **Uploaded media (Pro-only, private-only):** videos added from the user’s device and stored under the 2GB cap. Export includes **either** (1) media download links + manifest (default) **or** (2) media files in the zip if user explicitly requests and file size allows.

2.3 Deletion categories

Account deletion must remove all personal content and personal identifiers unless explicitly allowed for minimal retention. The default stance is: **delete everything user-specific** and retain only de-identified aggregates needed for analytics integrity.

- **Must delete:** profile, settings, authored moves/customizations, flows, practice logs, mastery/gameplans, maintenance plans, inbox items, share links, uploaded media, and any push notification tokens.
- **May retain (de-identified):** aggregated analytics counters (e.g., total number of exports per week) with no user_id linkage.
- **Must revoke:** share links created by the user so they no longer resolve.
- **Must not delete:** content that other users duplicated into their own libraries (imports saved as duplicates). Once duplicated, it belongs to the recipient’s account.

3. Data Export: Package, Formats, and Schema

Exports are delivered as a single .zip file. The zip is meant to be usable by: (a) a human reading JSON/CSV, and (b) a future Handz import tool (future work).

3.1 Export package structure (normative)

The generated zip **must** follow this directory layout so that exports are consistent over time.

Zip layout

```
handz_export__/
  manifest.json
  README.txt
  data/
    account.json
    profile.json
    settings.json
    moves.json
    flows.json
    flow_graphs.json
    sequences.json
    sharing_links.json
    inbox.json
    practice_sessions.json
    mastery_gameplans.json
    maintenance.json
    notes.json
  csv/
    practice_sessions.csv
    practice_sets.csv
    maintenance_tasks.csv
  media/
    media_manifest.json
    (optional media files or empty)
```

If a category has no items, include the file with an empty array rather than omitting it. This reduces guesswork for downstream consumers.

3.2 manifest.json (required)

The manifest is the single source of truth for what is inside the export, which schema versions were used, and how the export was generated.

```
{
  "export_id": "",
  "generated_at": "",
  "app": { "name": "Handz", "bundle_id": "HZ-V1", "export_schema_version": "1.0" },
  "user": { "user_id": "", "timezone": "", "plan_state": "free|trial|pro" },
  "counts": {
    "moves_total": 0,
    "flows_total": 0,
    "practice_sessions_total": 0,
    "gameplans_total": 0,
    "media_items_total": 0
  },
  "media": {
    "includes_media_files": false,
    "media_delivery": "links_only|embedded",
    "expires_at": ""
  },
  "integrity": {
    "sha256": { "data/moves.json": "", "data/flows.json": "" }
  }
}
```

User identifiers in exports are allowed because the user is downloading their own data, but the export must still be protected (reauth + rate limits; see §5). If we later support sharing export files, we should move to hashed identifiers by default.

3.3 JSON files: general rules

- All JSON must be UTF-8 encoded.
- Use arrays for collections (e.g., moves.json is an array of move objects).
- Include stable identifiers (canonical IDs) for moves and flows as defined in CH10/CH12.
- Timestamps must be ISO-8601 in UTC (e.g., 2026-01-02T23:59:59Z). If the UI stores local timestamps, export both local and UTC or include timezone in the record.
- Never export secrets (auth tokens, payment receipts, Apple transaction tokens).
- For privacy, include only the minimum profile fields needed for the user to recognize the account (display name, username, email if the user signed up via email; do not export password hashes).

3.4 CSV files: why and rules

CSV is provided for users who want to analyze training in spreadsheets. CSV files must be consistent and documented in README.txt.

- CSV delimiter: comma.
- Header row required.
- Escape quotes in cells and wrap cells containing commas/newlines in double-quotes.
- Include one CSV per major time-series: practice sessions, practice sets, maintenance tasks.
- Do not include nested JSON inside CSV cells unless absolutely necessary; if needed, include a parallel JSON file that contains the nested details.

3.5 README.txt (required)

README.txt is written for humans. It explains what is in the export, what the files mean, and how to interpret counts. It must also include the disclaimer that results vary for any training heuristics (owned by CH26).

4. Data Export UX (Screens, Buttons, and Copy)

This section defines the exact user journey for exporting data, including entry points, confirmations, progress states, and error states.

4.1 Entry points (must exist)

- **Settings Tab** → **Data & Privacy** section → **Export My Data**
- **Account** screen (if separate) → **Data & Privacy** → **Export My Data**

No other hidden entry points are required, but all entry points must route to the same export flow and use the same copy.

4.2 Screen list for export (owned by CH34)

- **Screen CH34-EX1:** Data & Privacy (Settings subsection). Contains Export and Delete actions.
- **Screen CH34-EX2:** Export Options (scope + include-media toggles).
- **Screen CH34-EX3:** Confirm Export (summary of what will be included).
- **Screen CH34-EX4:** Export Progress (generating zip; cancel; background behavior rules).
- **Screen CH34-EX5:** Export Ready (share sheet; save to Files; copy export_id).
- **Modal CH34-EXE:** Export Error (actionable message + retry).

4.3 Data & Privacy screen (CH34-EX1)

This is a subsection of Settings. It must be simple, not a wall of text.

Components:

- **Section header:** Data & Privacy
- **Row:** Export My Data (chevron) → routes to CH34-EX2
- **Row:** Delete Account (destructive styling) → routes to deletion flow (see §7)
- **Optional row:** Privacy Policy / Terms (owned by CH02/CH15 copy pack; include if app already has it)

Inline helper text (small): “Download a copy of your Handz data as a zip file (JSON + CSV).”

4.4 Export Options screen (CH34-EX2)

Goal: Let the user request an export without requiring them to understand schema details. Keep defaults safe.

Controls:

- **Export scope** (radio): (A) Everything (default) (B) Practice only (C) Flows and Moves only. (If we ship only Everything in V1, keep others hidden but reserve IDs.)
- **Include media** (toggle): “Include uploaded media” (default OFF). Subtext: “Uploads are private. Exporting media may create a large file.”
- **Delivery method** (radio): (A) Generate on device (default) (B) Email me a download link (optional, if we implement server-side). If only on-device in V1, show only A.

- **Button:** Continue → routes to CH34-EX3

Plan gating: Export is available to Free and Pro. Guest cannot export because Guest cannot save data (CH08). If Guest taps Export: show blocking prompt: “Create an account to export data.”

4.5 Confirm Export screen (CH34-EX3)

Content: A bullet summary of what will be included and an estimate of size and time. Must not promise exact times.

Copy requirements:

- Show counts if available: “2 flows, 18 moves, 9 practice sessions…”
- If include-media is OFF: “Uploaded media will not be included. Links will be included.”
- If include-media is ON: show warning callout: “This file could be large. Keep the app open during export.”
- Button (primary): Generate Export
- Button (secondary): Cancel

Upon tapping Generate Export, the app must require re-authentication if the last re-auth is older than the security window (see §5). Then route to CH34-EX4.

4.6 Export Progress screen (CH34-EX4)

This screen exists because exports can take time and can fail. It must provide clear progress and safe cancellation.

UI elements:

- Progress indicator: determinate if possible (0-100%) else step-based (Preparing data → Writing files → Finalizing).
- Text: “Generating your export…”
- Subtext: “Do not close the app.” (If background export is supported, modify copy accordingly.)
- Button: Cancel Export (secondary, confirm modal)
- If cancel: delete any partial export artifact and return to CH34-EX2 with toast: “Export canceled.”

Background behavior (V1 default): If the app goes to background, the export **may pause**. When resumed, it continues. If OS kills the app, export fails gracefully and user can retry. Do not attempt fragile background tasks unless we explicitly implement them.

4.7 Export Ready screen (CH34-EX5)

UI elements:

- Success state: “Export ready”
- Show export_id for support: “Export ID: XXXXX” with Copy button.
- Primary action: “Save to Files” (invokes iOS share sheet / file exporter).
- Secondary action: “Share…” (invokes share sheet; warn that the file contains personal data).

- Tertiary action: “Done” → returns to Settings.

If the export contains media download links with expiry, show: “Media links expire on .”

5. Data Export Security, Limits, and Abuse Controls

Exports are a high-risk feature because they can be used to exfiltrate data if an attacker gets temporary access to a device. The app must apply stronger safeguards here than for normal navigation.

5.1 Re-authentication rules (required)

- Before generating an export, require re-auth if the user has not re-authenticated within the last **10 minutes** (default).
- Re-auth methods must match the account method: Apple Sign-In prompt, Google reauth, or password entry for email accounts.
- If re-auth fails or is canceled: return to CH34-EX3 with non-blocking message: “Export canceled.”

Note: This is separate from unlocking Practice or other features. Export and Delete are treated as “sensitive actions.”

5.2 Rate limits (required)

Rate limits reduce abuse and accidental repeated exports. These are soft caps aligned with the product’s “not restrictive” philosophy.

- **Soft limit:** 3 exports per 24 hours per account. Above this, show a warning and require an extra confirmation.
- **Hard limit:** 10 exports per 24 hours per account. Above this, block and show: “For safety, exports are limited. Try again tomorrow.”
- If user hits limits, log an abuse event (owned by CH30) and display a “Why?” help link to a short explanation.

5.3 Export size safeguards

- If on-device export is selected, show an estimated size and warn if estimated size > 250MB.
- If include-media is ON and total media size > 1GB, show a blocking warning: “This may fail on your device. Consider exporting without media.” Provide two buttons: Export Without Media (default) and Continue Anyway.
- Never allow export to exceed the user’s 2GB stored media cap; exports should not attempt to embed more than stored.

5.4 Privacy warnings

When the user taps Share on CH34-EX5, show a one-time interstitial:

> “This export may contain personal data. Only share it with people you trust.”

Provide: [Cancel] [Continue]. Remember choice: user can disable warning in Settings (optional).

6. Export Generation Behavior (Deterministic Rules)

Even though implementation details vary, the behavior must be the same across builds. These rules define what the app must do, not which library to use.

6.1 Snapshot semantics

- An export is a snapshot at a point in time. The export must include a single timestamp (generated_at) and must not mix partial updates.
- If the user edits data during export, the app may either (A) lock editing until export completes, or (B) export the state at the start of export and allow editing. V1 recommendation: (B) allow editing; export is start-of-export snapshot.
- If snapshot fails due to sync conflict, show actionable error (see §11).

6.2 Ordering and determinism

- Within each JSON array, items must be sorted deterministically (e.g., created_at asc then id asc) so exports are diff-friendly.
- Do not randomize file names or order inside the zip; only export_id and timestamps vary.

6.3 Integrity checks

- Compute SHA-256 for each major file and include in manifest.json for integrity.
- If a file fails to write, treat the export as failed and do not produce a partial export for the user to share by accident.

6.4 Partial failures

If some non-critical file fails (e.g., one optional CSV), the app must either:

- **Fail the export** (recommended for V1) with a clear message: “Export failed while writing practice_sets.csv. Try again.”
- **Or** produce an export with a missing-file note in README and manifest warnings. (Only do this if we later observe frequent failures; not recommended in V1.)

7. Account Deletion UX (Screens, Confirmations, Copy)

Apple expects apps that support account creation to provide in-app deletion. Deletion must be explicit and safe.

7.1 Entry points (must exist)

- Settings → Data & Privacy → Delete Account (destructive row).
- Optional duplicate entry: Profile screen → Delete Account (if Profile exists).

7.2 Screen list for deletion (owned by CH34)

- **Screen CH34-DEL1:** Delete Account (information + consequences).
- **Screen CH34-DEL2:** Confirm Deletion (reauth + type-to-confirm).
- **Screen CH34-DEL3:** Deletion In Progress (blocking).
- **Screen CH34-DEL4:** Deleted (signed out; confirmation).
- **Modal CH34-DELE:** Deletion Error (retry / contact support).

7.3 Delete Account info screen (CH34-DEL1)

Goal: Explain consequences without overwhelming the user. This is where we prevent accidental deletion.

Content requirements:

- Title: “Delete account”
- Short line: “This permanently deletes your Handz data.” (No sugar-coating.)
- Bullet consequences (exact list below).
- Button (destructive): Continue to delete
- Button (secondary): Cancel

Consequences shown:

- Your saved flows, moves, notes, practice history, mastery/gameplans, and maintenance plans will be deleted.
- Your share links will be revoked and will stop working.
- Any flows you previously sent that other users duplicated will not be deleted from their accounts.
- If you have an active subscription, you must cancel it in iOS Settings. Deleting your account does not automatically cancel your Apple subscription.
- This action cannot be undone after completion.

7.4 Confirm deletion screen (CH34-DEL2)

This step requires re-authentication and an explicit confirmation gesture.

Controls:

- Re-auth prompt (same rules as exports; see §5.1).

- Type-to-confirm input: user must type **DELETE** (all caps) to enable final button. (Accessibility: allow case-insensitive match but display requirement as DELETE.)
- Final button (destructive, disabled until typed): “Delete my account”
- Secondary button: “Cancel”

Optional safety: Add a 3-second hold-to-confirm on the final button (press-and-hold). This reduces accidental taps. If implemented, write exact hold behavior in code and tests.

7.5 Deletion in progress screen (CH34-DEL3)

- Blocking screen with spinner: “Deleting your account…”
- Do not allow navigation away.
- If deletion takes longer than 10 seconds, show subtext: “This may take a moment. Keep the app open.”
- If app is backgrounded and deletion is interrupted, resume deletion on next launch or show a clear failure with next steps.

7.6 Deleted screen (CH34-DEL4)

- Confirmation: “Account deleted.”
- Button: “Return to Welcome” → routes to Welcome Gate (CH05/CH07).
- Optional: link to Privacy Policy.

8. Deletion Semantics, Retention, and What Happens to Each Data Type

Deletion must be implemented as a two-phase process to be reliable: (1) revoke access, (2) purge data. The user experience must feel immediate even if background purge takes time.

8.1 Recommended deletion model (V1)

- **Phase 1 (Immediate):** Mark account as “deleting” and sign user out. Revoke all share links immediately.
- **Phase 2 (Purge):** Within 24 hours (target) and no later than 30 days (hard requirement), delete all user content and media from storage.
- **No undo in V1:** Once deletion is initiated, do not offer an undo flow. (Undo adds complexity and legal ambiguity.)

If you later decide to add an undo grace period, that becomes a CH34 revision and must define exactly what is retained during the grace period.

8.2 Data-by-data deletion rules (normative)

The deletion worker must process items in a safe order so nothing is orphaned or left accessible. Order is listed below.

1. **Revoke share links** (so public access is cut off first).
2. **Invalidate sessions** / refresh tokens (so the user cannot keep using cached auth).
3. **Delete uploaded media** (largest cost; ensure storage cleanup).
4. **Delete flows and flow graphs** (including any references to media).
5. **Delete moves and move notes** (including links and metadata).
6. **Delete practice logs** (sessions, sets, summaries).
7. **Delete mastery/gameplans** and maintenance data.
8. **Delete inbox records** (received imports) and any pending conflict resolution records.
9. **Delete settings and notification tokens.**
10. **Delete user profile record.**
11. **Delete auth account** (Firebase Auth user / equivalent).

8.3 What is retained (minimal)

- Aggregated, de-identified analytics counters (owned by CH33). Must not contain user_id, email, username, or device identifiers.
- Security audit logs may retain a hashed user id for abuse investigations only if required; otherwise prefer aggregate-only. If retained, must be documented and justified.

8.4 Handling subscriptions on deletion

Because subscriptions are managed through StoreKit, deleting the account does not automatically cancel the subscription. The app must:

- Show a clear note on CH34-DEL1: “Cancel in iOS Settings.”
- Provide a button “How to cancel subscription” that opens the iOS subscription management deep link if feasible, otherwise opens instructions screen.
- After deletion, if the user re-installs later, they may still be subscribed at Apple level; entitlements are handled by CH25/CH08.

9. Interaction with Sharing, Inbox, Imports, and Conflicts

These interactions prevent contradictions and ensure that data export and deletion do not break the sharing funnel.

9.1 Export includes sharing objects

- Export must include all share links (active and revoked) and any metadata needed to understand them.
- Export must include inbox items (even if Free can only view them) so the user has a record of what they received.

9.2 Deletion and shared links

- All share links created by the user must be revoked immediately on deletion initiation.
- If someone opens an old link after revocation, they must see a safe public error page: "This link is no longer available." (Owned by CH17/CH31; referenced here.)

9.3 Deletion and imported duplicates

When a recipient imports a flow into their library, it becomes a duplicate owned by the recipient. Therefore:

- Deleting the sender account must not delete duplicates saved by recipients.
- If duplicates include the sender's custom move descriptions, those descriptions become part of the recipient's data and are not removed.
- Exports should reflect that ownership: imported flows in the user's library are exported normally as part of their content.

9.4 Deletion and conflict resolution artifacts

- If the user has pending import conflicts (CH19), those records must be deleted as part of deletion.
- If a conflict was resolved by creating new custom moves, those moves are also deleted.

10. Offline Behavior and Reliability Requirements

Exports and deletions must behave predictably under poor connectivity. Offline rules are owned by CH28; this section defines additional requirements specific to exports/deletions.

10.1 Export while offline

- If the export requires server fetch and the device is offline: block with message “Connect to the internet to export your data.”
- If the app maintains a fully local cache adequate for export, it may allow export offline. V1 recommendation: require connectivity to avoid incomplete exports.
- If connectivity drops mid-export: show error with retry. Retry must re-generate a fresh export (do not resume partial zip unless implemented safely).

10.2 Deletion while offline

- Deletion initiation requires connectivity (so we can revoke access server-side). If offline, block with: “Connect to the internet to delete your account.”
- If connectivity drops after initiation: the user may already be signed out; the server-side purge must complete when connectivity returns. On next login attempt, if account is flagged deleting, block login with message: “Account deletion in progress.”

10.3 Long-running operations

- If export or deletion exceeds 60 seconds, provide a persistent status message and keep the user informed.
- Do not leave the user stuck on a spinner indefinitely. Provide timeouts and clear next steps.

11. Error States (Export and Deletion)

CH31 owns global error state patterns. This section defines specific error cases and the required user-facing copy and actions.

11.1 Export errors (CH34-EXE modal)

- **No internet:** “You’re offline. Connect to the internet and try again.” Actions: [OK]
- **Re-auth canceled:** “Export canceled.” Actions: [OK]
- **Insufficient storage:** “Not enough space to create the export. Free up storage and try again.” Actions: [Retry] [Cancel]
- **Zip generation failed:** “Export failed while generating the file. Please try again.” Actions: [Retry] [Cancel]
- **Media export too large:** “Media makes this export too large for your device. Export without media instead.” Actions: [Export Without Media] [Cancel]
- **Unexpected:** “Something went wrong. Try again.” Actions: [Retry] [Cancel]

11.2 Deletion errors (CH34-DELE modal)

- **No internet:** “You’re offline. Connect to the internet to delete your account.” Actions: [OK]
- **Re-auth failed:** “We couldn’t confirm it’s you. Try again.” Actions: [Retry] [Cancel]
- **Server error:** “We couldn’t delete your account right now. Try again later.” Actions: [Retry] [Cancel]
- **Account already deleting:** “Account deletion is already in progress.” Actions: [OK]
- **Unexpected:** “Something went wrong. Try again.” Actions: [Retry] [Cancel]

11.3 Public errors for revoked links

Owned by CH17/CH31 but required here: revoked links must not reveal whether a user existed. Copy should be neutral: “This link is no longer available.”

12. Placeholder Registry (CH34-owned)

Anything not locked must remain a placeholder. When decided, update CH00 and bump CH34 revision as needed.

- **PLACEHOLDER:** Export scope options • Owner: CH34 • Options: Everything only (V1) / Add Practice-only / Add Flows-only • Default: Everything only • Decide-by: before build starts
- **PLACEHOLDER:** Export delivery method • Owner: CH34 • Options: On-device only / Email link / Both • Default: On-device only • Decide-by: before build starts
- **PLACEHOLDER:** Export rate limits • Owner: CH34 • Options: 3/day soft, 10/day hard (default) / Adjust after telemetry • Default: 3/10 • Decide-by: after CH33 metrics plan
- **PLACEHOLDER:** Media export behavior • Owner: CH34 • Options: Links-only (default) / Embed files / Both • Default: Links-only • Decide-by: after storage testing
- **PLACEHOLDER:** Deletion completion SLA • Owner: CH34 • Options: 24h target, 30d hard (default) / Faster • Default: 24h/30d • Decide-by: before launch

13. Acceptance Tests (Given/When/Then)

All tests below must pass for CH34 to be considered shippable. Write automated tests where possible; otherwise use QA scripts in CH35.

13.1 Export flow tests

- **Given** a logged-in Free user with 2 saved flows and 0 media uploads, **when** they navigate Settings → Data & Privacy → Export My Data and generate an export, **then** the app produces a zip containing manifest.json, README.txt, and all required data/*.json files with correct counts and deterministic ordering.
- **Given** a logged-in Pro user with uploaded media, **when** they leave Include media OFF, **then** the export contains media_manifest.json with links-only and does not include media files.
- **Given** a user who has not re-authenticated within 10 minutes, **when** they tap Generate Export, **then** a re-auth prompt appears and export does not start until re-auth succeeds.
- **Given** the user cancels re-auth, **when** they return to the app, **then** the export is canceled and no export file is created.
- **Given** the app is backgrounded during export, **when** the user returns, **then** the export either continues or fails gracefully with a clear retry path; it must not silently succeed with partial files.
- **Given** the user taps Cancel Export during generation, **when** they confirm, **then** partial export artifacts are deleted and the user returns to Export Options with toast “Export canceled.”
- **Given** the user hits the hard export limit, **when** they attempt another export, **then** the app blocks with the limit message and does not start export.

13.2 Deletion flow tests

- **Given** a logged-in user, **when** they navigate to Delete Account and proceed, **then** they see the consequences list and must continue to confirmation.
- **Given** the user has not re-authenticated within 10 minutes, **when** they proceed to confirm deletion, **then** re-auth is required.
- **Given** the user has not typed DELETE, **when** they view the confirm screen, **then** the final delete button is disabled.
- **Given** the user types DELETE and confirms, **when** deletion begins, **then** the user is signed out and sees Deletion In Progress; share links are revoked immediately.
- **Given** a revoked share link, **when** someone opens it, **then** they see “This link is no longer available.”
- **Given** the deleted user re-installs the app, **when** they attempt to log in, **then** login is blocked (or fails) because the account no longer exists.
- **Given** recipients duplicated the user’s flows previously, **when** deletion completes, **then** recipient duplicates remain accessible.

14. Replit Build Prompt (Implement CH34 Only)

Paste the prompt below into Replit Agent along with CH00 and CH34 PDFs. This prompt instructs the agent to implement only the export/deletion features and to respect cross-references.

```
You are implementing Handz V1, chapter CH34 only (Data Export & Account Deletion).
You MUST follow the PRD bundle rules from CH00: no detail loss, required front-matter
behaviors, and cross-references.
Do not implement unrelated features.

Context:
- iOS only, portrait only.
- Auth: Firebase Auth (Apple/Google/Email) per CH07.
- Data exists in Firestore/Storage per CH29.
- Plan states: Guest/Free/Trial/Pro per CH08.
- Practice is paywalled and Free has 3 monthly practice credits; Free has 2 saved flows;
Free inbox cap 10 (CH00 locks).

Tasks:
1) Add Settings → Data & Privacy subsection (CH34-EX1) with rows: Export My Data, Delete
Account.
2) Implement Export flow screens CH34-EX2..EX5, including:
- options (scope=Everything only for V1; include-media toggle default OFF)
- confirm screen with counts
- re-auth gating (10-min window)
- progress screen with cancel
- zip generation on-device: create folder layout, manifest.json, README.txt, data/*.json,
csv/*.csv, media_manifest.json
- deterministic ordering and sha256 integrity hashes
- rate limits (soft 3/day, hard 10/day)
- share sheet to Save to Files; and warning interstitial before Share
3) Implement Delete Account flow screens CH34-DEL1..DEL4:
- consequences list and subscription cancel note
- re-auth + type DELETE confirm
- revoke share links immediately
- mark account deleting, sign out, then purge user data and storage items via secure
backend callable function (recommended) or server-side worker
- ensure duplicates imported by other users are not deleted
4) Implement error handling per CH34 §11 and wire to global error pattern per CH31.

Constraints:
- If you must make an assumption, write it into a PRD Assumptions comment block and stop
that part until resolved.
- Do not create large UI tables. Use vertical lists and cards.
- Provide unit tests or QA scripts for the Given/When/Then in CH34 §13.

Deliverables:
- Source code changes implementing CH34 screens and logic.
- A short checklist mapping each PRD requirement to the code location.
- A troubleshooting note list for common failures (zip creation, re-auth, storage cleanup).
```

15. Troubleshooting Notes (CH34-specific)

This section is written for you (or Replit Agent) when something goes wrong during implementation or QA.

- **Export zip is missing files:** Ensure the generator always writes empty arrays for empty categories and that file creation errors fail the export (CH34 §6.4).
- **Export ordering differs between runs:** Verify deterministic sort keys (created_at asc then id asc) are applied before writing JSON arrays.
- **Export crashes on large data:** Stream writes instead of holding entire dataset in memory; disable include-media by default; warn above 250MB.
- **Re-auth never triggers:** Confirm the app tracks lastSensitiveAuthAt timestamp and checks the 10-minute window before Generate Export / Delete Account.
- **Deletion seems to succeed but data remains:** Verify the purge worker deletes Storage objects and Firestore docs; confirm share links are revoked first; ensure auth account deletion is last step.
- **Users still access revoked links:** Confirm link resolver checks revoked flag or existence; do not rely on cached data; set cache headers if using web endpoints.
- **Subscription confusion after deletion:** Ensure the deletion screens include the explicit note and the “How to cancel subscription” button; do not claim cancellation happens automatically.
- **Offline edge cases:** Block initiation when offline; show actionable copy; resume purge server-side when possible.

End of CH34.