PRACTICECOACHVERIFIED 2026-07-10 · CSEA RETENTION ADDED 2026-09-04 · MODERATION REVIEW HOLD ADDED 2026-09-06 · END-OF-HOLD CORRECTED 2026-09-07 · ERASURE CORRECTED 2026-10-12 · ACCOUNT-HOLD RELEASE ADDED 2026-10-12 · OWNER BLOCK ADDED 2026-10-31 · PLANS AND PAYMENTS ADDED 2026-11-07 · TRIAL CLAIMS AND WEBHOOK LOG ADDED 2026-11-08

FOR YOUR DPO / DATA-PROTECTION LEAD

Data Retention Statement

What PracticeCoach keeps, for how long, and what is not automated — stated honestly

Data Retention Statement

Describes PracticeCoach's *current* retention behaviour as of July 2026. Prepared for DPO review — not legal advice. Numbers below are the concrete figures the shipped tooling actually honours; where something is not automated, this statement says so plainly rather than imply a guarantee that does not exist.

Principle

PracticeCoach retains personal data only for as long as it is needed to provide the service, and gives users self-serve control over their own data. This statement is deliberately **hybrid**: it quotes exact numbers only where the shipped product enforces them, and is explicit about the gaps.

Two exceptions to the principle, and both are set out below rather than left to contradict the sentence above. *The first is statutory, not a business choice:* where a report has been made to the National Crime Agency under the CSEA reporting duty, reg. 8 of SI 2026/268 requires specific material to be **kept** for a fixed period (§3a). *The second is a safety control, and it is new as of 6 September 2026:* content removed by moderation is held for a bounded review period so that a wrong removal can be undone, and is then deleted (§3b). **The two end differently, and this sentence was corrected on 7 September 2026 because it previously implied they did not.** The review hold is bounded and ends in an **automatic** deletion by an hourly job. The reg. 8 retention is a statutory **minimum** rather than a window: nothing deletes held material automatically — the hourly job excludes held rows by construction — and the hold ends by a deliberate operator step recorded in `content-moderation-run-book.md` §5 step 6. That is the position already filed in `online-safety-risk-assessment.md` §5 (ICU D10), and it is deliberate: an unattended deletion of criminal-category evidence is the one deletion in this service that must have a human on it.

---

1. Deletion requests — 14-day reversible grace, then erasure within a day

When a user requests deletion of their account:

- The account is **flagged for deletion with a 14-day reversible grace period.** Nothing is destroyed immediately; the user (or the school, on the user's behalf) can reverse the request within that window with **Undo** on the same screen. *(Mechanism: a `pending_deletion_at` timestamp on the user record.)*

- The account is erased **within a day of the grace period ending, by a scheduled job. Corrected 14 October 2026:** this used to say the account is erased "*shortly after the grace period ends, by a job that runs hourly*". The job is scheduled hourly, but the scheduler (GitHub Actions) in practice starts it only every four to eight hours, so "hourly" over-promised. The figure is now **within a day**; moving the job to a database-side scheduler is planned. Every flagged account whose grace has elapsed goes through **one erasure routine** — the same routine an operator uses to erase an account from the admin screen — which deletes, in this order: the files under the user's storage prefixes (uploaded and imported sheet music, recordings, message attachments), the user's calendar-feed files, and any ordinary (non-statutory) moderation-quarantine objects of theirs; then the **sign-in account itself**, whose deletion removes every row keyed to the user by database cascade. The routine then counts what is left keyed to the user and fails loudly if anything is. *(Mechanism:* `eraseAccount` *in* `api/_lib/accountErasure.ts`*, called by the scheduled notifications job,* `api/cron/notifications.ts`*; the cascades and the residue count are migration 206.)* **What alerts if a run is missed (added 14 October 2026):** a run that starts and fails (an error, or a refused secret) fails the GitHub Actions workflow, and GitHub reports the failure to the repository owner. A run that GitHub never starts raises **no alert**: nothing yet watches for a missing run, so "within a day" is the normal case, not a monitored guarantee. Each run is also capped at 60 seconds, so a long backlog is worked through over several runs.
- **A failed attempt is retried, never dropped — up to the point the account itself is deleted.** If any step before that fails, the deletion flag is left set, the error is logged with its reason, and the next scheduled run (normally within a day) tries again. **Corrected 12 October 2026 (pre-ship review, IN-01):** this bullet used to say "*If any step fails, the deletion flag is left set … and the next hourly run tries again*" without the qualification. That is not true of the last two steps. Once the sign-in account has been deleted the flag has gone with it, so a failure in the final re-sweep of the user's files, or a residue count that is not zero, **cannot be retried by the job**: it is reported as an error to the operator (Sentry, with the list of what was left), and what is left is found and removed with `scripts/sweep-erasure-residue.mts`. The same script is the backstop for one known gap: a device that was already signed in when the erasure ran can go on uploading until its session expires (about an hour), and those files are orphaned in the erased user's storage folder until that script is run. So the honest figure is **14-day reversible grace → erasure by a scheduled job, normally within a day of it ending**; no particular hour is promised. **(Aligned 14 October 2026:** this sentence used to say "*erasure shortly after it ends*", which the corrected bullet above had already replaced with "within a day".)

- **Some accounts are refused, not erased**, and are handled by an operator instead (four cases that apply to every route, and a fifth and a sixth that apply to the hourly job only): an **operator account** (one that has taken moderation actions or granted guardian-gate overrides — the audit trail must keep saying who did what, so the account is retired by hand), an account that still holds **admin access** (revoked first), an account **under a CSEA statutory hold** (§3a) — erasure is refused until an operator has both ended the hold on every held file and recorded that the hold on the account is released (`content-moderation-runbook.md` §5 steps 6 and 7), after which the erasure proceeds as for any other account — and an account **whose sheet-music folder holds pages that belong to other accounts** (**added 12 October 2026**: erasing it would delete files another person's music still points at, so the routine refuses and reports how many pages are involved until an operator has moved them. No ordinary account is in this state — every route that gives one person another's music copies the file into the recipient's own folder — and on the day this was written the only examples were seeded demonstration and test accounts). **Added 12 October 2026 (second pre-ship review, WR2-02): a fifth case, which applies to the hourly job only.** If the account is **suspended** when its grace period ends, the hourly job does not erase it. A suspension is the first step an operator takes when a report about an account is serious, and the statutory hold above begins only later, when the material is held under a legal rule; in between, a deletion request the person had made *before* they were suspended would have fallen due and the job would have destroyed, with nobody deciding to, the very files the review is about, including material already taken down. So the request **waits until the review is finished**. An operator then either lifts the suspension, after which the hourly job erases the account on its next run, or erases the account from the admin screen, which is not refused for a suspended account because there a person is deciding. This changes when an in-app deletion request completes, and the privacy policy and the delete-account page say so. **Added 31 October 2026 (Phase 144): a sixth case, which also applies to the hourly job only.** If the account **owns a studio, department or ensemble that has other members** when its grace period ends, the hourly job does not erase it, because erasing the owner would delete the group, and with it data other members still use. The app already refuses to schedule a deletion for such an owner (the Account page tells them to hand the group over to another member or close it first), so this case arises only where members joined during the grace period, or the request was made before 31 October 2026. The request waits, and is logged and alerted once a day like the fifth case, until the owner hands the group over or closes it, or cancels the request. An operator's erasure from the admin screen is not refused for an owner, because there a person is deciding; it still deletes the group for its members (see below). A refused account keeps its deletion flag and is logged on every hourly run, so it cannot be silently dropped. (**Changed 12 October 2026:** an account under a statutory hold, and a suspended account waiting on a review, are still logged and counted on every run, and are raised to the operator as an alert once a day rather than on every run, because neither is something the operator can end at once.) **Added 12 October 2026 (second pre-ship review, WR2-03):** the refusal for an account under a statutory hold is enforced by the application's erasure routes (the hourly job and the operator's erasure from the admin screen) and not by the database, so deleting that account's user directly in the Supabase

dashboard or with a script is not refused; operators are instructed never to delete such a user directly (`content-moderation-runbook.md` §5), and enforcement in the database itself is planned.

Corrected 12 October 2026. This section used to read: "*After the grace period, an **hourly scheduled job hard-purges** the flagged accounts. (Mechanism: an hourly cron job that permanently removes records whose grace period has elapsed.)*" and *"So the honest, enforced figure is: **14-day reversible grace → hourly hard-purge.**"* That was not true. The job deleted the user's practice data but **never deleted the sign-in account**, so the profile (email, display name, date of birth), push subscriptions, studio memberships, messages, feedback and stored files survived, and the person could still sign in. It is quoted rather than deleted so the change is auditable. As of Phase 104 the account itself is erased, as described above.

What survives an erasure, stated rather than left to be inferred:

- **Complaints the person filed survive, de-identified.** A `content_reports` row they submitted is kept as part of the Online Safety Act complaints record, with the reporter removed.
- **The moderation record survives, de-identified.** `moderation_actions` rows about the person are kept with the affected-user link removed. When an operator erases an account from the admin screen, the audit row for that erasure keeps a **one-way SHA-256 fingerprint of the account's email address** (trimmed and lower-cased), so that a later complaint ("you deleted my account") can be matched without keeping the address. A fingerprint of an email address is **pseudonymous personal data**, not anonymous data; it is kept with the moderation record and for as long as that record is.
- **Bare user ids remain on other people's rows**, with no other personal data attached: the target id on a moderation action, the rights-attestation id on a received copy of a score (the rights provenance of a copy someone else now owns), and the owner id on a moderation-quarantine ledger row. They point at an account that no longer exists.
- **Guardians' copies of notices are kept.** Where a notice about a child was sent to their parent or guardian, the guardian's sent-notification record (including the rendered email) belongs to the guardian's account and is kept as the guardian's own record.

What goes with the account that a reader might not expect:

- A studio, department or ensemble the person **owns** is deleted with the account, with its memberships, works and shares. **Since 31 October 2026** an owner cannot schedule deletion in the app while the group has other members (the sixth case above), so in an in-app deletion this happens only to a group with no other members; it still happens to a group with members when an operator erases the owner from the admin screen. Members keep music they already imported into their own accounts (received copies survive the deletion of the share they came from).
- Message threads the person was part of are deleted, **including the other party's view of them**.

---

2. Teacher-authored records — deleted with the pupil's account

When a pupil's account is erased, records the **teacher** authored about that pupil — progress reports, and the teacher's annotation layers on the pupil's pieces — are **deleted with the pupil's account** (progress reports cascade from the pupil; teacher layers hang off the pupil's pieces, which go). When a **teacher's** account is erased, the teacher's own progress reports are deleted first, so none is left without an author.

There is no automatic "pupil left a teacher" anonymisation flow today. A database helper that redacts a teacher's records about a pupil exists (migration 110) and is service-role-only, but nothing calls it; it is kept for a possible future flow when a teacher–pupil link ends.

Corrected 12 October 2026. This section used to read: "*When a pupil leaves or their link to a teacher ends, records the **teacher** authored about the pupil (progress-report notes, annotations) are **anonymised rather than destroyed.** … This "anonymise, don't destroy" posture is a deliberate design choice, not an accident of the data model.*" No flow performed that when a link ended; the only caller was the account-deletion job, which Phase 104 replaced with erasure. Deletion is the stronger outcome for the pupil, and this section now says what happens. It is quoted rather than deleted so the change is auditable.

---

3. Self-serve export

Users can **export their own data at any time** from the in-app privacy settings (`/settings/privacy`). The export is a downloadable archive of the user's practice record (pieces, passages, sessions, confidence log, events, plan history, run-throughs) and **now includes message history** as well. This is the primary mechanism by which the school can meet a data-subject access or portability request for a pupil, and by which teacher↔pupil message history is **exportable / disclosable** where DMs have been enabled.

3a. Statutory retention — CSEA reports to the National Crime Agency

Added 4 September 2026. This is the one place in the product where a legal duty requires us to **keep** something rather than to delete it, and it is recorded here because the principle above — "retains personal data only for as long as it is needed to provide the service" — does not describe it and would be untrue without this section.

Section 66 of the Online Safety Act 2023, commenced for user-to-user services by the Online Safety (CSEA Content Reporting by Regulated User-to-User Service Providers) Regulations 2026 (SI 2026/268, in force 7 April 2026), requires this service to report detected and unreported child sexual exploitation and abuse content to the National Crime Agency. **Once such a report has been made, reg. 8 imposes a retention duty:**

What is kept	For how long	From
The unique NCA report reference number	**Five years**	The date of the report
The reported content itself, the information submitted, whatever was used to judge it CSEA, and the uploader's **associated data** (account information, upload and login IP addresses and any port numbers with their dates and times, and — per reg. 8(2) — a fortnight of account connections)	**One year**	The date of the report

Three things about this that a DPO should be told plainly:

- **It applies only where a report has been made, and it is not the same thing as the review hold in §3b. Corrected 6 September 2026.** This bullet used to read: "*No routine retention of removed material exists, no quarantine bucket exists, and none is being described here. Ordinary takedowns are still hard deletes, and a successful appeal is answered by asking the user to re-upload.*" That is no longer true — §3b describes the review hold that replaced it — and the sentence is quoted rather than deleted so the change is auditable. What remains true, and is the point of this bullet: reg. 8 is not a licence to keep material generally; it is a duty to keep *specific* material, for a bounded period, once a report has been made. Material held on the reg. 8 clocks is segregated from the ordinary review hold, is never restored to the account it came from, and is not deleted by the sweep that enforces §3b **at all** — that sweep excludes held rows by construction. **Corrected 7 September 2026:** this bullet used to say "not deleted early by the sweep", which read as though the sweep deleted it eventually. It does not. The one-year date in the table above is the **earliest** an operator may end the hold, following `content-moderation-runbook.md` §5 step 6, which now sets out that step and its order; it is not a date on which anything happens by itself.
- **Material converted to a statutory hold while its owner is being erased is not destroyed. Added 12 October 2026 (WR2-02).** An ordinary removal can be converted to a hold at any time. If that happens while the owner's account is part-way through an erasure, the routine re-reads each removed item immediately before deleting its file, does not delete one that has become a hold, and stops before the account itself is deleted; the account is then under the hold like any other. One limit is stated plainly: a conversion landing in the few milliseconds between that re-read and the deletion of the file cannot be prevented, and in that case the routine stops and reports it instead of recording the item as dealt with.

- **A statutory hold is a refusal to erase, and it is lifted by two recorded operator steps.** Where a report was made, an erasure request for the uploader's account — whether from the user's own deletion after its grace period or from an operator — is **refused**, so the account data reg. 8 requires is kept. That refusal is made by the application's erasure routes and not by the database: a direct deletion of the user in the Supabase dashboard or by script is not refused, operators are instructed never to delete such a user directly (`content-moderation-runbook.md` §5), and enforcement in the database itself is planned (**added 12 October 2026**, second pre-ship review, WR2-03). The refusal ends only when **both** of these have happened, and neither is ever done by a job: (1) an operator has **ended the hold on every held file**, after the earliest date and on the NCA's and the police's confirmation that they do not need the content (`content-moderation-runbook.md` §5 step 6); and (2) an operator has **recorded that the hold on the account is released**, on their separate confirmation that they do not need the account (§5 step 7). The second step is recorded with the operator, the time and a stated reason, in a table only the server can read or write and in the moderation record. Recording it deletes nothing; **the deletion then proceeds by the ordinary routes** — the hourly job, on its next run, if the person's own deletion request is pending and due, or an operator's erasure. A file held after a release puts the account back under the hold until that file's hold has ended and a new release is recorded. **Corrected 12 October 2026 (pre-ship review, WR-04):** this bullet used to read "*is refused while the account has a statutory-hold record, so the account data reg. 8 requires is kept. No job ends the hold or erases a held account: both are deliberate operator steps under `content-moderation-runbook.md` §5.*" Two things in it were wrong. The statutory-hold record is never deleted, so "while the account has" one meant for ever: no step, by an operator or otherwise, could lift the refusal, although the privacy policy says the account is deleted once the hold ends. And the runbook's own table said the step that ends the content hold also ended the account's, which disagreed with this statement and with the runbook's step 7. Both documents now describe the same two steps, and the second one exists. It is still true that **no job ends a hold**; what a job may now do is complete a person's own pending deletion request **after** an operator has recorded the release. **Corrected 12 October 2026** — this bullet used to read: "*Where the retention is longer than the ordinary clock, it governs. Account metadata is otherwise kept 90 days after deletion in case the police ask for it; where a report was made, reg. 8's one-year clock is longer and is the one that applies.*" No such 90-day hold exists, and nothing ever implemented it: outside a statutory hold, an erased account's metadata is deleted with it (§1). The sentence is quoted rather than deleted so the change is auditable.

- **The provisions are now verified against the primary text — with one thing still open.** Regs. 3(4), 4(1), 6(2), 6(6)–(7), Sch. 1 and Sch. 2 were verified against the SI when this section was written; **reg. 8, reg. 8(2) and reg. 4(6) were read at `legislation.gov.uk/uksi/2026/268/regulation/8/made` and `…/regulation/4/made` on 6 September 2026 and are quoted in full in `online-safety-risk-assessment.md` §6.** The clocks above are confirmed correct as filed. **Still open, and not discharged by the quotation:** the NCA must be asked at registration how it wants the content itself held, and asked again at report time; that is an operational answer a regulation cannot supply. Note also that reg. 8(2)'s "relevant data" is wider than the file — two weeks of the uploader's files, messages, chat logs and connection attempts prior to the upload — and that the tooling preserves the object and its metadata only; the rest is a manual preservation step recorded in `content-moderation-runbook.md` §5 step 2.

Procedure: `content-moderation-runbook.md` §5, order of operations, step 6. **Assessment record:** `online-safety-risk-assessment.md` §4 and §6.

3b. Moderation review hold — removed content is kept for 30 days, then deleted

Added 6 September 2026, and it changes a sentence this statement previously carried. Until this release, moderation removals were hard deletes and this document said so. They are not any more.

When an operator removes content on a report — uploaded pages, a copied score's pages, a studio resource, a message recording, an ensemble announcement, or the text of a passage note — the file is **moved to a private bucket under a fresh key** rather than deleted, and the row that named it is kept with a timestamp and the reason given. The URL the file was served from stops resolving. The purpose is narrow and it is a safety one: a removal made in error can be reversed, so that a false positive does not cost a musician their score permanently. Reference copies and bundle copies are ordinary copies for every verb in this list: a copy taken *for reference* (kept outside the planner) or delivered as part of a bundle is removed, held, restored and purged exactly as any other copied score is.

What is kept	For how long	From	What happens at the end
The removed file, in a private bucket not reachable by any public URL, plus a ledger row naming the content, its owner, the reason and the report it answered	**30 days by default** (an operator setting, `app_settings.moderation_review_window_days`, changeable without a deploy — if it is changed, this figure changes with it)	The moment of removal	An hourly job deletes the file and the rows, permanently, and writes its own audit record. An operator can also delete it earlier, as a second explicit action.

Five things a DPO should be told plainly:

- **It is not an archive and it is not browsable.** There is no gallery. An operator sees a list row — what, whose, why, when, and the clock — and can open the file itself only through a link that is minted per click, expires in minutes, and is logged.
- **The end of the window is enforced by a job, not by a promise.** Deletion happens on the hourly schedule whether or not anyone looks at the queue.
- **Restoring puts the file back under a new URL**, never the old one, and the ledger records both the removal and the restore.
- **The user is told when their content is removed** (they already were) and is emailed if it is put back. They are not told the length of the window, and no user-facing surface exposes the queue.
- **Content held under the CSEA path in §3a is not part of this**: it is excluded from the review queue's restorable list, is never restored, and runs on the reg. 8 clocks instead.

Procedure: `content-moderation-runbook.md` §4 and §5. **Assessment record:** `online-safety-risk-assessment.md` §2.3, §5 (ICU D10) and §6. **Privacy policy: discharged 7 September 2026.** This line used to read *"this retention must be named in the policy — an outstanding item owned by Phase 85 (COMP-04)"*, and it is quoted rather than replaced so the item can be seen to have been closed rather than dropped. `src/pages/PrivacyPolicy.tsx`, under **Security and retention**, now states the 30-day hold, the statutory exception in §3a and the fact that moderation actions are recorded — in user voice, with the same figures as this section. If the operator setting is ever changed, the policy sentence is one of the two places that has to move with it.

3c. Ensemble readiness consent — kept while the ensemble and the account exist, not for a period

Added 15 September 2026, with the v4.1 ensemble readiness work. A member of an ensemble can choose to share **how they are getting on with that ensemble's music** — their confidence and tempo on the pieces shared there, and nothing else. The choice is recorded as a row, and this section states how long that row lives, because it is the one retention claim in this document that is scoped to the life of two objects rather than to a clock.

The row. One row per explicit grant, in `ensemble_readiness_consents` (migration 185 §B), with six columns and no more: `id`, `studio_id` (the ensemble), `member_id` (the person), `status` — one of `accepted` or `revoked` — `consented_at` and `revoked_at`. There is no copy of the readings in it: the readiness view is computed on request from the member's own practice data, so the consent row records the **decision**, never the data the decision released.

What is kept	For how long	What ends it
The consent row — the ensemble, the member, `accepted` / `revoked`, and the two timestamps	**While the ensemble and the account both exist.** Not a fixed period, and not an indefinite audit record that outlives either.	Deleting the **ensemble** or the **account** removes it: both foreign keys are `ON DELETE CASCADE` (to `studios` and to `auth.users`), so it is a **hard delete** by the database, not a job that has to run.

Leaving is a soft revoke, not a deletion. When a member leaves an ensemble or is removed from it, a database trigger (`trigger_sweep_readiness_consents`, migration 185 §E) flips the row to `revoked` and stamps `revoked_at`. The sharing stops at that moment — the projection reads live consent only — and the row survives as the record of what was shared and when it stopped. Because the unique index that enforces one live consent per (ensemble, member) exempts revoked rows, a **re-grant is a new row rather than an overwrite**, and a member who re-joins the ensemble shares **nothing** until they choose again. Revocation is terminal for the row it applies to.

Why the distinction is stated rather than summarised. The easy sentence — "consent records are kept as an audit trail" — is not true here, and an earlier draft of migration 185's own commentary said something close to it and was corrected under review (**WR-07**). A retention claim a DPO relies on that overstates retention is a UK GDPR storage-limitation misstatement, not a wording preference. The posture this section describes is the same one every other row keyed to a studio or to `auth.users` has, and it is what the account-deletion promise in the privacy policy requires: if a member deletes their account, nothing about their consent to share survives it.

If a future phase needs a severance record to outlive the ensemble it belonged to, the answer is a retention table written before the cascade — not a weaker foreign-key action, which would leave rows pointing at a studio that no longer exists.

Schema: `supabase/migrations/185_readiness_flag_and_consent.sql` §B (the table, the indexes and the cascade) and §E (the sweep trigger and the retention paragraph as WR-07 corrected it). **Privacy policy:** `src/pages/PrivacyPolicy.tsx`, under **Security and retention** — the same three facts in user voice: kept while the ensemble and your account exist, leaving switches it off, deleting either deletes it outright. The sharing itself is described under **Who can see your data**. **DPIA:** *DPIA Information Pack* §1 (the purpose), §2 (the closed column list) and §6 (this retention, in summary).

3d. Session recordings — 30 days at PostHog, then deleted

Added 28 October 2026 (#529). An adult who uses PracticeCoach on their own may opt in to session recordings: a replay of taps, scrolls and screens, with typed input, notes, images and recordings hidden, and messages, chat, studio, family and ensemble screens never recorded. Accounts that are under 18, of unknown age, a pupil of any teacher, a guardian or in a school department are never asked (`src/lib/sessionReplay.ts`). The recordings are held by **PostHog, EU cloud (Frankfurt)**, keyed to the internal user id only, under the DPA signed 28 October 2026 (`posthog-dpa-2026-10-28.pdf`). Client IP addresses are discarded by the project setting.

- **Retention is PostHog's project setting, kept at 30 days.** Recordings expire there automatically; nothing in PracticeCoach deletes them. If the project setting is changed, the privacy policy's "30 days" must change with it.
- **Account erasure does not reach them.** The erasure routine in §1 does not call PostHog, so a deleted account's recordings remain until they expire, at most 30 days. The privacy policy says so.
- **Withdrawal stops recording at once** (Privacy & Data → Session recordings) but does not delete recordings already made; they expire on the same clock. A person can ask us to delete them sooner, which is done by hand in PostHog.
- **Not in the export.** The self-serve export (§3) does not include recordings.

3e. Usage events and sign-up admissions — kept for the life of the account

Added 5 November 2026 (Phase 161). Two first-party tables, both in Supabase EU, both keyed to the account and both removed by the database when the account is erased.

What is kept	For how long	What ends it
`usage_events` — one row per tracked action (page views, client errors, admin breadcrumbs, and since Phase 161 the `firstrun.*` setting-up steps and `home.*` card and prompt events). Short enum properties only: no titles, notes or typed content.	**For the life of the account.** There is no fixed purge period today.	Erasure of the account (§1): the foreign key to `auth.users` is `ON DELETE CASCADE`, so the rows are hard-deleted by the database.
`signup_admissions` — one row per admitted account: the user id, the admission route and the time. Server-only.	**For the life of the account.**	The same cascade on `auth.users`.

- **Neither is in the export or a personal backup** (§3). `usage_events` is insert-only for the client and cannot be read back by it; `signup_admissions` is server-only.
- **The sign-up details themselves are not kept here.** The date of birth, invite or join code and Terms version a person enters at sign-up are held in the sign-in record's metadata only until the profile is created, then scrubbed (DPIA Information Pack §10). The date of birth is kept on the profile, as before. If no profile is ever created (a refused or never-confirmed sign-up), the hourly job clears those details from the sign-in record once it is seven days old; the sign-in record itself, with its name and email address, is not deleted.
- **Honest gap.** A fixed retention period for `usage_events` (for example, ageing out rows older than a set number of months) has not been set. If one is introduced, this table and the privacy policy change with it.

3f. Plans and payments — payment records for 6 years, the rest with the account

Added 7 November 2026 (Phase 175); `trial_claims` and `billing_webhook_events` filled in 8 November 2026 (Phase 176); `stripe_customers` and the guardian-payer link added 8 November 2026 (Phase 177). The plan and payment tables built in phases 169, 173, 174 and 176, all in Supabase EU. The DPIA Information Pack §11 describes each one; this is how long each is kept. A school's pupils and staff get the full plan through the school's invoice, so for a school deployment most of these tables hold only the plan record and, where a pupil answered the optional sign-up questions, their attribution and survey answer.

What is kept	For how long	What ends it
`payment_events` — the financial record of each payment, refund, chargeback and reversal (amounts, tax, fees, period, provider reference, partner). Server-only, append-only.	**6 years from the end of the financial year it relates to**, the period UK company law requires for accounting records.	Account erasure (§1) does **not** delete it: `user_id` is set to null (`ON DELETE SET NULL`), so the record stays with the person removed. Deletion at the end of the 6 years is operator step today (see the honest gap belov
`studio_access_endings` — the studio's name, the end date and grace end when studio access ends. Server-only.	**30 days** from the end date.	The hourly downgrade sweep (`free_tier_prune_access_endings` , migration 260), or account erasure (`ON DELETE CASCADE`), whichever comes first.
`downgrade_notices` — what the person was told about a plan ending and when, and for an applied downgrade which pieces moved to their archive. Server-only.	**Until account deletion.** Applied rows are the person's record of what moved.	Account erasure (`ON DELETE CASCADE`).
`entitlements` (the plan record, including an admin grant's reason and admin ids), `downgrade_choices` , `user_attribution` , `signup_survey` , `partner_code_redemptions` .	**For the life of the account.** "Start over" also clears `downgrade_choices` .	Account erasure (`ON DELETE CASCADE`).
`trial_claims` — one row per free trial claimed (migration 263): the channel, a one-way SHA-256 hash of the Stripe card fingerprint for a web trial or the store's original transaction id, the provider reference and the claim time. Server-only.	**3 years from the claim.** It outlives account deletion so one person cannot take a second trial.	Account erasure (§1) does **not** delete it: `user_id` is set to null (`ON DELETE SET NULL`). The nightly billing reconcile (`api/_lib/cron/billingReconcile.t` once a day from 03:00 UTC on the hourly cro deletes rows 3 years after `claimed_at` .
`billing_webhook_events` — one row per message from Stripe or RevenueCat (migration 262): provider, event id and type, environment, subject reference, account id, status, error text, and the payload with names, emails, addresses and all RevenueCat subscriber attributes except `partner_code` and `prompt` removed. Server-only.	**Payload: until processed**, at most **30 days** for a failed event. **Event row: 90 days.**	The webhook handler deletes the payload on the event is processed; the nightly billing reconcile nulls a failed event's payload after 3 days and deletes event rows after 90 days. Account erasure sets `user_id` to null (`ON DELETE SET NULL`). If Stripe cannot be reached when an account is erased (or the gr refund check cannot run), the erasure still go ahead and writes one extra row here (type `pc.erasure_cancel` , no account id) naming only the Stripe customer and subscription ids still to cancel or to check for renewal refund, and the time deletion was requested; the nightly reconcile retries it and deletes its payload once the subscription is cancelled, and the same 30-day and 90-day clocks apply.
`stripe_customers` — the Stripe customer reference for an account that has paid on the web (migration 268, Phase 177): account id, Stripe customer id, live or test. Server-only.	**For the life of the account.**	Account erasure (`ON DELETE CASCADE`), after the erasure job has read it to cancel any active Stripe subscription. Stripe keeps its ow financial records under its own terms.

What is kept	For how long	What ends it
`entitlements.payer_user_id` — on a child's plan record, which account pays for it (migration 268, Phase 177).	**While both accounts exist.**	Deleting the child's account removes the plan record (`ON DELETE CASCADE`); deleting t paying account sets the column to null (`ON DELETE SET NULL`), and the child's subscription is set to cancel at the end of the paid period, so the plan lasts until then.

- **None of these is in the export or a personal backup** (§3): a restore must never grant a new account a plan or attribute it to a partner.
- **On the device.** `pc_partner_code` (localStorage) is kept for **30 days**; `pc_pending_signup_heard_about` (sessionStorage) only until the account exists. Neither leaves the device except as part of creating the account.
- **Admin ids on another person's plan record.** If the admin who granted, changed or ended a full plan deletes their own account, those ids are set to null on the plan record; the record stays.
- **Honest gap.** Stripe and RevenueCat keep their own records of payments and subscriptions under their own privacy policies and legal duties, as do Apple and Google for in-app purchases. Erasing a PracticeCoach account does not delete those. There is no scheduled job yet that deletes `payment_events` rows once their 6 years have passed; until there is, it is an operator step.

---

4. Backups — provider-managed only

Database backups are **managed by our infrastructure provider** (Supabase) via point-in-time recovery according to our plan tier. **We do not assert a specific backup-retention day-count in this document** — the exact recovery window depends on the live plan and should be **confirmed in the Supabase dashboard at signing** rather than quoted from a figure that may be stale. Backups exist for disaster recovery; they are not a separate user-facing store and are subject to the provider's own security terms.

---

5. Honest gaps — what is *not* automated

In the interest of not overstating the tooling:

- **No inactivity-based auto-deletion.** PracticeCoach does **not** currently auto-delete accounts purely because they have been dormant for a period. Data persists until the user (or the school, on written instruction) requests deletion.

- **Closing a studio or department removes its guardian-gate override records (known gap, recorded 31 October 2026).** Since 31 October 2026 an owner can close a studio or department from its settings. Closing deletes the studio, its memberships, student links, settings, lessons and payment records. It is refused while any of the studio's library material is under a moderation hold. Closing also removes any `guardian_gate_overrides` rows recorded against the studio, because they cascade with it. Those rows are part of the operator audit trail. Deleting an ensemble already removed them in the same way before this date. The grant and the revocation of each override are still recorded in `moderation_actions`, which survives. The cascade has not been changed.
- **End-of-contract delete-or-return is on the school's written instruction, and is currently manual.** At the end of a school deployment, the Art. 28(3)(g) delete-or-return obligation is carried out **on the school's written instruction.** Bulk school offboarding is today a **manual service-role operation** — there is no self-serve "delete the whole department" button. (A department owner can close the department itself since 31 October 2026, which ends its classes and department consents for everyone, but that does not delete any pupil's or teacher's own account or data.) We will not imply automation that does not exist; dedicated bulk-offboarding tooling is a deferred item.

---

6. Cross-references

- **Sub-processors, regions and the data-flow map:** see the *DPIA Information Pack*.
- **The contractual retention and delete-or-return obligations:** see the *DPA / Processor Terms* template, clauses 3(e), 3(f) and 3(g).
- **The statutory CSEA retention clocks and the procedure behind them:** see `content-moderation-runbook.md` §5 step 6 and `online-safety-risk-assessment.md` §4 and §6.
- **The 30-day moderation review hold:** §3b above; `content-moderation-runbook.md` §4 and §5; `online-safety-risk-assessment.md` §2.3 and §5 (ICU D10).
- **What the user is told about both clocks:** `src/pages/PrivacyPolicy.tsx`, *Security and retention* — the same 30 days, the same statutory exception, and the moderation log, in plain user voice (added 7 September 2026). The DPO-facing summary is *DPIA Information Pack* §6.
- **Plans and payments:** §3f above; *DPIA Information Pack* §11 (each table, its grants and the providers) and the §5 footnote that RevenueCat and Stripe are not engaged for schools; `src/pages/PrivacyPolicy.tsx`, *Security and retention* (the 6 years, the 30 days, the device keys) (added 7 November 2026), and the 3-year trial record and the Stripe and RevenueCat messages (added 8 November 2026).

This statement describes PracticeCoach's current retention behaviour as of July 2026, amended 4 September 2026 to add the statutory CSEA retention duty in §3a, amended 6 September 2026 to add the moderation review hold in §3b (and to withdraw the "ordinary takedowns are still hard deletes" sentence it replaces), and amended 7 September 2026 to correct how the reg. 8 hold ENDS: by an operator step in the runbook, never by the hourly job, which excludes held rows by construction. Amended 12 October 2026 (Phase 104, pre-ship review WR-04) so that a statutory hold on an account can end: it is lifted by two recorded operator steps (the hold on each held file ended, then the hold on the account recorded as released), after which a pending deletion proceeds. Amended 12 October 2026 (Phase 104, second pre-ship review WR2-02) so that the hourly job does not erase a suspended account — a person's own deletion request waits until the review of a report about the account is finished — and so that material converted to a hold during an erasure is not destroyed. Amended 12 October 2026 (Phase 104) to correct §1, §2 and §3a: the deletion job now erases the sign-in account and its files rather than only the practice data; teacher-authored records about a pupil are deleted with the pupil's account, and no "pupil left a teacher" flow exists; the 90-day post-deletion metadata hold never existed and is struck; and what survives an erasure (de-identified complaints and moderation record, the email fingerprint, bare ids, guardians' copies) is named. Amended 14 October 2026 to say erasure happens within a day of the grace period ending rather than by a job that "runs hourly" (§1). Amended 5 November 2026 (Phase 161) to add §3e, `usage_events` *and* `signup_admissions`*. Amended 7 November 2026 (Phase 175) to add §3f, plans and payments, and its §6 cross-reference. Amended 8 November 2026 (Phase 176) to set the retention of* `trial_claims` *(3 years, kept past deletion with the person removed) and add* `billing_webhook_events` *(payload until processed, at most 30 days on failure; event rows 90 days) in §3f, both enforced by the nightly billing reconcile. It is provided for DPO review and is not legal advice. Behaviour verified against the shipped system 2026-07-10; the backup-retention window and any provider terms should be re-confirmed at signing. The reg. 8 clocks were re-read against the primary text on 6 September 2026 and are quoted in* `online-safety-risk-assessment.md` *§6; the NCA's own answer on how it wants content held is still to be obtained at registration.*