

CH17 — Sharing: Unlisted Links + Link Lifecycle (R1)

Generated: 2026-01-02 • Bundle ID: HZ-V1

Doc ID:	HZ-V1-CH17_Sharing_Unlisted_Links_Link_Lifecycle_R1
Revision:	R1 (2026-01-02)
Status:	Draft
Depends on:	HZ-V1-CH04 (Navigation Map), HZ-V1-CH07 (Auth), HZ-V1-CH08 (Entitlements), HZ-V1-CH16 (Flow API), HZ-V1-CH17 (Placeholder Registry).
Related:	HZ-V1-CH18 (Inbox), HZ-V1-CH19 (Import Conflicts), HZ-V1-CH25 (Paywalls), HZ-V1-CH28 (Offline Sync).
Supersedes:	—
Owned Decisions:	Unlisted sharing model; link lifecycle/state machine; link management UI; share link viewer mode rules.
Open Questions / Placeholder Items:	See CH31 (Placeholder Registry).

1. Purpose and Scope

This chapter specifies the complete Version 1 sharing system for Handz: creating **unlisted share links** for a saved flow, opening/viewing those links, managing link lifecycle (copy/share/revoke/optional expiry), and all edge cases required to make sharing reliable and abuse-resistant.

In scope:

- Create an unlisted share link for a saved flow (owner action).
- Open a share link (recipient action) and view the flow in **read-only viewer mode**.
- Link lifecycle: Active → Revoked (and optional Expired/Disabled states).
- Link management UI: list/copy/share/revoke and minimal link stats.
- Entitlement-aware restrictions (Guest vs Free vs Pro vs Trial) as they apply specifically to link creation and link management (see CH08 for global gating).
- Error states and recovery paths specific to sharing (see CH31 for global patterns).

Out of scope (owned elsewhere):

- Inbox storage, import caps, and save-to-library mechanics (See: HZ-V1-CH18).
- Conflict resolution when imported flows contain missing/custom moves (See: HZ-V1-CH19).
- Practice paywall/credits rules and practice on shared items (See: HZ-V1-CH20–CH22 and HZ-V1-CH25).
- Full navigation map and screen inventory (See: HZ-V1-CH04, HZ-V1-CH05).

- Deep offline/sync conflict rules (See: HZ-V1-CH28).
- Warning ladder, hard abuse actions, and enforcement messaging templates (See: HZ-V1-CH30).
- Analytics event list + privacy policy language (See: HZ-V1-CH33).

2. Definitions and Terms

Core glossary terms are owned by CH03. This chapter defines only sharing-specific terms and states.

- **Unlisted Link:** A share link that is not discoverable/browsable in-app and is only accessible to people who have the URL.
- **Link Token:** The unguessable identifier embedded in the URL. Must not be sequential. The server treats it as a secret.
- **Link Record:** The backend record mapping token → flow snapshot reference and metadata (owner, status, timestamps).
- **Viewer Mode:** A read-only flow view rendered for recipients. Editing is disabled; import actions route to Inbox (CH18).
- **Link Lifecycle:** State transitions for a link: Created → Active → (Revoked | Expired | Disabled).
- **Active Link:** A link whose status allows viewing. It may still be subject to plan gates (e.g., import requires account).
- **Revoked Link:** A link the owner has intentionally disabled. It must not open the flow.
- **Expired Link (optional V1):** A link that becomes invalid after a set time.
- **Disabled Link:** A link disabled by the system for abuse/safety reasons (See: CH30).

3. Product Goals for Sharing (V1)

- **Growth loop without communities:** Sharing is the V1 viral loop: people receive a flow → view it → create an account → import → become invested.
- **Zero confusion:** Recipients must understand what they're looking at (a flow/gameplan) even if they've never seen a flow builder before.
- **Safety and abuse resilience:** Links must be unguessable; creation must have rate limits and soft caps; revocation must work instantly.
- **Low-friction for coaches and training partners:** Sharing must be 2 taps from the Flow Detail screen and feel instant.
- **Trust:** The sender's flow renders identically for the recipient (within V1 constraints, especially around video attachments; see §9.6).

Non-goals (V1):

- Public feed / browseable directory of user flows.
- Follow systems, profiles, or community discovery.

- Real-time collaborative editing.
- Sharing entire libraries as a first-class feature (explicitly deferred).

4. Entry Points and Surfaces

This section specifies where sharing actions appear in the UI, and what the user can do at each entry point.

4.1 Primary entry point: Flow Detail View

- Location: Flow Detail View (See: HZ-V1-CH16).
- Primary actions: **Share, Copy Link, Manage Links** (owner only).
- Visibility: only for saved flows (flows must be owned/saved; guests cannot save flows by V1 rules).

4.2 Secondary entry point: Library list quick actions (optional V1)

- Location: My Library list item overflow menu (See: HZ-V1-CH15).
- If included: Offer link to Flow Detail's Share surface (avoid duplicating logic).
- If excluded: All sharing remains within Flow Detail to reduce complexity.

4.3 Recipient entry point: Opening a share link

- Recipient taps URL in Messages/DM/Notes/etc.
- If app installed: app opens directly into **Share Link Viewer**.
- If app not installed: show a minimal landing page (or App Store redirect) that explains what Handz is and prompts install. (Deep-link transport decision owned by CH28; see placeholders §13).

5. Entitlement Rules (Guest/Free/Pro/Trial)

Global plan definitions and gating rules are owned by CH08 and monetization specifics by CH25. This chapter defines only the share-link-specific behavior.

5.1 Who can create share links

- **Guest:** Cannot create share links (guests cannot save flows; See: CH07/CH08).
- **Free:** Can create unlisted share links for saved flows, subject to soft caps and abuse limits.
- **Pro/Trial:** Same as Free but with higher caps and fewer friction warnings.

5.2 Who can open share links

- **Anyone with the link** can open and view in Viewer Mode, but import/save actions may require account and may be capped (See: CH18).
- If the recipient is a Guest, they can view but cannot save/import until they create an account (conversion prompt required; See: CH07).

5.3 Share link counts and caps (V1)

Exact caps are not locked in CH00 yet and must be treated as placeholders. This chapter specifies the mechanics and warning behavior; the final numbers are decided in §13.

Cap types (all should be implemented as soft caps with warnings first; See: CH30):

- **Daily create cap:** number of new links the owner can create per rolling 24h.
- **Active link cap:** number of Active links allowed per owner at one time.
- **Per-flow cap:** optional limit on active links per flow (prevents spam).

6. Link Lifecycle and State Machine

A share link is a durable artifact that can be revoked. The system must be deterministic: the same token must always resolve to the same outcome given its status.

6.1 States

- **CREATED:** Record exists but not yet visible to recipient (rare; typically transitions immediately to ACTIVE).
- **ACTIVE:** Link opens successfully into Viewer Mode.
- **REVOKEDED:** Owner revoked; link must not reveal flow content.
- **EXPIRED (optional V1):** Link is time-expired. Treat similarly to REVOKEDED for UX.
- **DISABLED:** System-disabled due to abuse/safety enforcement (See: CH30).

6.2 Transition rules

State transitions must be server-authoritative. Client caches are never trusted for link validity.

Allowed transitions:

```
CREATED → ACTIVE → (REVOKEDED | EXPIRED | DISABLED) ACTIVE → REVOKEDED (owner action)  
ACTIVE → DISABLED (system action) ACTIVE → EXPIRED (time) DISABLED → ACTIVE (rare;  
admin/unban) REVOKEDED → ACTIVE (optional: only if owner explicitly 'Re-enable';  
default V1: no, create a new link)
```

6.3 Revocation semantics

- Revocation must be immediate: any subsequent open attempts show the 'Link not available' screen.
- Revoking a link must not delete the underlying flow.
- Revocation is irreversible in default V1: to share again, create a new link (prevents accidental reactivation of a widely-leaked URL).

6.4 Deleting or renaming the underlying flow

- If the owner deletes the flow: all links to that flow become invalid and should display 'This flow is no longer available.'
- If the owner renames the flow: link remains valid; viewer should display the updated name.
- If the owner changes internal content: link behavior depends on snapshot policy (see §7.4).

7. Data Model (Share Link Records)

This section defines the minimum data needed to support link creation, opening, management, and safety. Implementation specifics are for Replit prompts at the end, but the model is defined here to avoid guesswork.

7.1 Share Link Record fields (minimum)

```
share_link - id: UUID (internal primary key) - token: string (secret token; store hashed server-side) - owner_user_id: UUID - flow_id: UUID (owner's flow) - status: enum('ACTIVE', 'REVOOKED', 'EXPIRED', 'DISABLED') - created_at: timestamp - revoked_at: timestamp? (nullable) - expires_at: timestamp? (nullable; optional V1) - last_opened_at: timestamp? (nullable) - open_count: int - last_opened_ip_hash: string? (nullable, privacy-aware) - last_opened_user_agent_hash: string? (nullable) - abuse_flag: bool (default false) - notes: string? (internal; admin/moderation; optional)
```

7.2 Token requirements

- Token must be unguessable: minimum 128 bits of entropy (recommend 160–192 bits).
- Token must not encode user_id or flow_id directly.
- Do not use incremental IDs in URLs.
- Server should store only a hash of the token (e.g., SHA-256) and compare hashed tokens.

7.3 URL format

Exact domain/path may change, but the app must treat the token as opaque.

<https://handz.app/s/>

Alternate acceptable formats:

<handz://share/> <https://handz.app/share/>

7.4 Snapshot policy (critical)

When a link opens, does the recipient see the **current** version of the flow or the **snapshot** at time of sharing?

Default V1 decision: show the **current** flow version.

- Pros: coach updates flow and students always see latest.
- Cons: link content can change after being shared; less reproducible.
- Mitigation: show 'Last updated' timestamp in viewer header.

If later we adopt snapshotting, it becomes a CH19/CH29 storage decision (larger storage and conflict implications).

8. Owner UX: Create, Copy, Share, Manage, Revoke

This section defines exact UI behavior and copy for link creation and management. Layout specifics (fonts/colors) are owned by CH06, but the interaction rules and copy are defined here.

8.1 Flow Detail: Share primary action

Placement (reference): Flow Detail top-right or primary action row. (See: HZ-V1-CH16).

Buttons (owner view):

- **Share** (opens Share Sheet modal/screen)
- **Copy Link** (one-tap create-or-copy)
- **Manage Links** (opens link list for this flow)

8.2 Copy Link behavior (one-tap)

- If there is an existing ACTIVE link for this flow and the user has not requested a new link: copy the most-recent ACTIVE link.
- If no ACTIVE link exists: create a new ACTIVE link, then copy.
- If user is over cap: show cap warning and give a path to manage/revoke old links.

Toast/feedback copy:

- Success (existing): “Link copied”
- Success (created): “New link created and copied”
- Failure (offline): “No connection - try again”
- Failure (cap): “Link limit reached - revoke one to create a new link”

8.3 Share Sheet screen (recommended)

A dedicated share screen reduces confusion and provides room for education and safety copy.

Screen title: Share Flow

Content blocks:

- Flow card summary (name, last updated, optional tags).
- Primary button: “Copy link”
- Secondary: “Share...” (invokes OS share sheet after link is ready)
- Link safety note: “Anyone with this link can view this flow.”
- Video note (if flow contains uploaded videos): “Uploaded videos aren’t included in shared links. Links to videos are shareable.” (See: CH29).
- Link management entry: “Manage links”

8.4 Manage Links screen (per flow)

Purpose: Show existing links for this flow and allow revoke/copy/share.

List item fields:

- Status pill: ACTIVE / REVOKED / EXPIRED / DISABLED
- Created date/time (local)
- Optional: “Opened X times” (if we surface)
- Actions: Copy • Share... • Revoke (ACTIVE only)

Empty state: “No links yet. Create one to share this flow.”

9. Recipient UX: Link Opening and Viewer Mode

This section defines what happens when someone opens an unlisted share link.

9.1 Routing outcomes

- If token resolves to ACTIVE: open Viewer Mode flow view.
- If token resolves to REVOKED/EXPIRED: open ‘Link not available’ screen.
- If token resolves to DISABLED: open ‘Link not available’ screen with different copy (avoid accusing user; See: CH30).
- If token invalid / not found: open ‘Link not found’ screen.

9.2 Viewer Mode screen requirements

- Must visually communicate read-only mode (e.g., “Viewer mode” badge).
- Flow content must be viewable with pan/zoom as needed (See: CH12).
- Editing controls are hidden or disabled.
- Primary CTA is entitlement-aware and owned by CH18 (Inbox). This chapter only defines the entry points:
 - - If logged in: show “Save to Inbox” (or “Import”) button.
 - - If not logged in: show “Create account to save” button.

9.3 Minimal educational copy (because flows are novel)

Shown near the top of Viewer Mode, collapsed by default after first view.

- **Header line:** “This is a Handz flow - a map of what to throw next based on what your opponent does.”
- **Subtext:** “Save it to your Inbox to practice or edit later.” (Practice gating handled elsewhere.)
- **Dismiss:** “Got it” and “Don’t show again”

9.4 Import/save actions (owned elsewhere)

When recipient taps Save/Import, route into Inbox import flow (See: HZ-V1-CH18) and then conflict resolution if needed (See: HZ-V1-CH19).

9.5 Sharing onward (recipient reshare)

- Recipients can forward the same unlisted URL. That is expected behavior.
- If recipient saves/imports into their own library, they may create their own new share links from their saved copy (subject to their plan caps).

9.6 Media attachment rule (critical transparency)

V1 decision from CH00/CH29 direction:

- Uploaded videos are private-only, count toward 2GB cap, and are **not** included in share links.
- Links (URLs) to externally hosted videos **are** shareable (because they are just text fields).
- Viewer Mode must show a clear badge when a node contains a private upload: “Video not shared (private upload).”

10. Copy Pack (V1)

Exact strings used in sharing flows. Keep copy short, confident, and non-technical. Tone guidance owned by CH15, but these strings are required for implementation.

- **Share sheet title:** “Share flow”
- **Share sheet note:** “Anyone with this link can view this flow.”
- **Copy link button:** “Copy link”
- **Share button:** “Share...”
- **Manage links:** “Manage links”
- **Copied toast:** “Link copied”
- **Created+copied toast:** “New link created and copied”
- **Cap reached:** “Link limit reached - revoke one to create a new link”
- **Revoked confirmation title:** “Revoke this link?”
- **Revoked confirmation body:** “Anyone who has it won’t be able to open this flow anymore.”
- **Revoked confirm CTA:** “Revoke link”
- **Revoked success:** “Link revoked”
- **Link not found title:** “Link not found”
- **Link not found body:** “This link doesn’t exist or was typed wrong.”
- **Link not available title:** “Link not available”
- **Link not available body (revoked/expired):** “This link was revoked or expired.”
- **Link disabled body:** “This link is no longer available.”
- **Viewer badge:** “Viewer mode”
- **Viewer education header:** “Flows map what to throw next.”
- **Viewer education body:** “Save to your Inbox to practice or edit later.”
- **Guest CTA:** “Create an account to save this flow”

11. Error Handling and Edge Cases

Global error-state layout patterns (empty states, blocking vs non-blocking) are owned by CH31. This chapter lists the required sharing-specific cases.

11.1 Owner-side errors

- No network when creating link: keep user on share sheet, show retry.
- Link created but copy/share fails (OS clipboard/share sheet error): keep link ACTIVE and show “Couldn’t copy - try again”.

- Cap reached: show cap message and route to Manage Links. Provide quick ‘Revoke oldest link’ option only if user confirms.

11.2 Recipient-side errors

- Token not found: show Link not found screen; provide “Paste a different link” (optional) or “Go to Home”.
- Revoked/expired: show Link not available screen; provide “Ask sender for a new link” suggestion line.
- Content removed (flow deleted): show “This flow is no longer available.”
- Open in-app fails because app not installed: landing fallback handled by deep link transport (CH28 placeholder).

11.3 Data consistency and caching

- Viewer Mode must always validate link status from server on open (do not trust local cache).
- If server validation fails due to timeout: show skeleton/loading then “Try again”.
- If link becomes revoked while viewer is open: allow viewing session to continue, but any refresh/import action should revalidate and fail gracefully.

12. Abuse, Safety, and Rate Limiting (Share-Specific)

This chapter defines the surface-level mechanics; the warning ladder and enforcement language is owned by CH30.

12.1 Required anti-abuse mechanics

- Rate limit link creation (rolling window) to reduce spam.
- Rate limit link opens from same IP/device to reduce scraping.
- Flag suspicious patterns (many creates, many opens, high failure rates) and set abuse_flag for review.
- System must be able to DISABLE a link and/or temporarily block link creation for an account (See: CH30).

12.2 UX philosophy (avoid punishing legit users)

- Use **soft caps** first (warnings), then block only at high thresholds.
- When blocking, provide next best action: manage/revoke links, or try later.
- Never show accusatory copy. Keep it neutral.

13. Placeholder Registry (Decide Later)

These items must be locked later and then reflected back into CH00 Decision Log.

- **PLACEHOLDER: SHARE_DAILY_CREATE_CAP_FREE** • Owner: CH17 • Options: 5 / 10 / 20 per 24h • Default: 10 • Decide-by: before beta
- **PLACEHOLDER: SHARE_DAILY_CREATE_CAP_PRO** • Owner: CH17 • Options: 25 / 50 / 100 per 24h • Default: 50 • Decide-by: before beta
- **PLACEHOLDER: SHARE_ACTIVE_LINK_CAP_FREE** • Owner: CH17 • Options: 10 / 25 / 50 active • Default: 25 • Decide-by: before beta
- **PLACEHOLDER: SHARE_ACTIVE_LINK_CAP_PRO** • Owner: CH17 • Options: 100 / 250 / 500 active • Default: 250 • Decide-by: before beta
- **PLACEHOLDER: SHARE_PER_FLOW_ACTIVE_CAP** • Owner: CH17 • Options: none / 3 / 5 • Default: 5 • Decide-by: before beta
- **PLACEHOLDER: LINK_EXPIRY_SUPPORT** • Owner: CH17 • Options: Off / 7 days / 30 days • Default: Off • Decide-by: beta
- **PLACEHOLDER: DEEP_LINK_TRANSPORT** • Owner: CH28 • Options: Universal Links / (Alt) dynamic links provider • Default: Universal Links • Decide-by: alpha
- **PLACEHOLDER: LINK_STATS_VISIBILITY** • Owner: CH17 • Options: show opens / hide opens • Default: show opens only to owner • Decide-by: beta

14. Acceptance Tests (Chapter-Specific)

Write tests as Given/When/Then. These tests must pass for CH17 to be considered complete.

- **Create link (no existing):** Given an authenticated user with a saved flow and no active link, when they tap Copy link, then a new ACTIVE link is created and copied, and a toast says “New link created and copied”.
- **Copy existing link:** Given an authenticated user with an ACTIVE link for the flow, when they tap Copy link, then no new link is created, and the existing link is copied, and a toast says “Link copied”.
- **Revoke link:** Given an ACTIVE link, when owner revokes it and confirms, then link status becomes REVOKED, the list updates immediately, and toast says “Link revoked”.
- **Open active link:** Given an ACTIVE link token, when a recipient opens it, then Viewer Mode renders the flow and shows a “Viewer mode” badge.
- **Open revoked link:** Given a REVOKED link token, when opened, then “Link not available” screen appears with body “This link was revoked or expired.”
- **Token not found:** Given a random token, when opened, then “Link not found” screen appears.
- **Uploaded video transparency:** Given a shared flow node containing a private upload, when recipient views it, then a label indicates the video is not shared.
- **Cap reached:** Given owner is at active link cap, when they attempt to create a new link, then app blocks creation, shows “Link limit reached - revoke one to create a new link”, and provides navigation to Manage Links.
- **Offline create attempt:** Given device is offline, when owner taps Share... then Copy link, then app shows “No connection - try again” and no link is created locally.
- **Server timeout on open:** Given link validation times out, when recipient opens the link, then app shows loading for a reasonable time, then a retry state without crashing.

Checklist

- Share UI present on Flow Detail.
- Copy link creates-or-copies as specified.
- Manage Links list works and reflects status changes instantly.
- Viewer Mode renders for active links and blocks for revoked/invalid.
- All copy strings match §10.
- Caps implemented as soft caps with warnings (CH30 alignment).
- No sensitive identifiers appear in URLs.

15. Replit Build Prompt (Implement CH17 Only)

Paste this into Replit Agent along with CH00 and CH17. It is written to prevent guessing and to keep implementation scoped.

You are implementing Handz V1. Follow CH00 rules strictly (no guessing; cross-reference dependencies).

Implement ONLY: HZ-V1-CH17 (Sharing: Unlisted Links + Link Lifecycle).

Dependencies you must assume already exist or stub:

- Auth (CH07) and plan states (CH08)
- Flow Detail View (CH16) and flow storage (flow_id exists)
- Inbox import flow exists as a route (CH18) – do NOT implement inbox logic here

Build:

1) Data model:

- Create a share_links table/collection with the fields specified in CH17 §7.1.
- Token must be unguessable and stored hashed server-side. URLs must include only the token (CH17 §7.2-7.3).

2) Flow Detail UI:

- Add actions: Share, Copy link, Manage links (CH17 §8.1).
- Copy link is create-or-copy behavior (CH17 §8.2).
- Share opens Share Sheet screen with required blocks and copy (CH17 §8.3 + §10).

3) Manage Links screen:

- List links for the flow with status pills and actions Copy/Share/Revoke (CH17 §8.4).
- Revoke changes status to REVOKED immediately and invalidates future opens (CH17 §6.3).

4) Link opening:

- Register route /s/<token> (universal link handler stub is ok; but in-app token open must work).
- On open: validate token with server and route:
- ACTIVE → Viewer Mode
- REVOKED/EXPIRED/DISABLED → Link not available screen
- NOT FOUND → Link not found screen
- Viewer Mode is read-only and shows "Viewer mode" badge (CH17 §9.2). Provide a CTA button that routes to Inbox import route (CH18) if logged in; otherwise to signup (CH07). Do not implement the import itself.

5) Error handling:

- Implement owner-side offline create error, cap reached error, and recipient-side invalid/revoked screens exactly as CH17 §11.
- Use CH31 error patterns if available; otherwise create minimal compliant screens.

6) Caps:

- Implement placeholder caps as configurable constants (CH17 §13). Show cap message when exceeded.
- Treat caps as soft caps first; if blocked, route to Manage Links.

Deliverables:

- Screens: ShareSheetScreen, ManageLinksScreen, ShareLinkViewerScreen (read-only), LinkNotFoundScreen, LinkNotAvailableScreen

- Unit-level tests or simple QA scripts aligned with CH17 §14
- A short README describing how to manually test: create → copy → open → revoke → open again

16. Troubleshooting Notes (CH17)

Use this when something feels ‘off’ during implementation or QA.

16.1 Common failures and fixes

- **Links open sometimes, fail other times:** likely caching or client-side validation. Fix: always validate link status server-side on open (CH17 §11.3).
- **Revoked link still opens:** server isn’t enforcing status. Fix: enforce status in the token lookup endpoint; do not rely on client.
- **‘Copy link’ keeps making new links:** missing ‘reuse existing active link’ rule. Fix: query most recent ACTIVE link for that flow before creating.
- **Recipients can edit flow:** Viewer Mode not truly read-only. Fix: disable edit controls and block edit routes while in viewer state.
- **Share sheet shows wrong copy:** ensure strings match §10 exactly (case and punctuation).
- **Cap errors feel too restrictive:** increase placeholder caps, but keep anti-abuse rules in CH30.

16.2 Diagnostic checklist

- Confirm token entropy: not sequential; appears random.
- Confirm server stores token hashed; raw token not stored.
- Confirm revoked/invalid tokens return correct screen.
- Confirm viewer CTA routes correctly based on auth state.
- Confirm uploads are marked “not shared” in viewer when applicable (CH29 alignment).

End of CH17