<https://gemini.google.com/u/1/app/28b5eb1f3acf0522?is_sa=1&android-min-version=301356232&ios-min-version=322.0&campaign_id=bkws&utm_source=google&utm_medium=cpc&utm_campaign=2024enUS_gemfeb&pt=9008&mt=8&ct=p-growth-sem-bkws>

<https://gemini.google.com/u/1/app/39ab8052602be0ce>

<https://gemini.google.com/u/1/app/8f9ebd64aa0fd57f>

### 3\. Updated Production-Ready Specification: `public.user\_waypoint\_short\_tips`

Version: 2.1.1 (Internal revision post-checklist review)

Date: May 18, 2025

1\. Purpose & Primary Use-Cases

- Mission Statement: To store brief, user-submitted textual tips, observations, or mini-reviews about specific waypoints, subject to a moderation workflow, to provide practical, recent, and nuanced insights for fellow pilgrims.

- Key User-Story Touchpoints:

- Pilgrim (Anna) - Story A8: Allows Anna to read valuable, concise tips from other pilgrims and to contribute her own tips, enhancing community engagement and shared knowledge.

- Host (Marco): Indirectly, hosts can gain feedback on their establishments or services through tips left by pilgrims.

- Regional Manager (Sofia): Can monitor tips for real-time insights into trail conditions, POI accuracy, or emerging issues/highlights within their managed region.

- Admin Team - Story D1: Directly supports content management through the comprehensive moderation workflow for tips, ensuring quality and appropriateness. Admins can also feature important tips using `is\_pinned\_by\_admin`.

2\. Schema

| column | data\_type | constraints | description |

| `id` | `bigint` | `PRIMARY KEY`, `GENERATED ALWAYS AS IDENTITY` | Unique identifier for each tip. |

| `profile\_id` | `uuid` | `NOT NULL`, `FOREIGN KEY to public.profiles(id) ON DELETE CASCADE` | The ID from the `profiles` table of the user who submitted the tip. If the user's profile is deleted, their tips are also removed. |

| `waypoint\_id` | `integer` | `NOT NULL`, `FOREIGN KEY to public.waypoints(id) ON DELETE CASCADE` | The ID from the `waypoints` table that this tip is about. If the waypoint is deleted, associated tips are also removed. Crucial: Confirm type matches `waypoints.id`. |

| `tip\_text` | `text` | `NOT NULL`, `CHECK (char\_length(tip\_text) > 0 AND char\_length(tip\_text) <= 500)` | The actual content of the tip. Character limit (e.g., 500 characters) enforced by DB constraint. |

| `language\_code` | `text` | `NOT NULL`, `FOREIGN KEY to public.languages\_master(language\_code) ON DELETE RESTRICT` | The language in which the tip was written by the user (e.g., 'en', 'it'). Links to active languages defined in `languages\_master`. |

| `tip\_category\_code` | `text` | `NULLABLE`, `FOREIGN KEY to public.tip\_categories\_master(category\_code) ON DELETE SET NULL ON UPDATE CASCADE` | Optional category code for the tip from `tip\_categories\_master`. If category is deleted, this becomes NULL. |

| `tip\_source\_platform` | `text` | `NULLABLE` | Optional: Indicates the platform from which the tip was submitted (e.g., 'web\_app\_v1', 'mobile\_ios\_v1'). Useful for analytics. |

| `moderation\_status` | `public.content\_moderation\_status\_enum` | `NOT NULL`, `DEFAULT 'pending\_approval'` | The current moderation status of the tip (e.g., 'pending\_approval', 'approved\_visible', 'rejected\_hidden'). |

| `is\_publicly\_visible` | `boolean` | `NOT NULL`, `GENERATED ALWAYS AS (moderation\_status = 'approved\_visible'::public.content\_moderation\_status\_enum AND deleted\_at IS NULL) STORED` | Computed column: True if tip is approved and not soft-deleted. Requires PostgreSQL 12+. |

| `is\_pinned\_by\_admin` | `boolean` | `NOT NULL`, `DEFAULT false` | If true, administrators have marked this tip as particularly important or helpful. |

| `moderated\_by\_profile\_id` | `uuid` | `NULLABLE`, `FOREIGN KEY to public.profiles(id) ON DELETE SET NULL` | The ID from `profiles` of the admin/moderator who last took a moderation action on this tip. |

| `moderation\_timestamp` | `timestamp with time zone` | `NULLABLE` | Timestamp of when the last moderation action was taken. |

| `moderation\_notes\_internal` | `text` | `NULLABLE` | Internal notes from the moderator regarding their decision. Not typically shown to general users. |

| `created\_at` | `timestamp with time zone` | `NOT NULL`, `DEFAULT now()` | Timestamp of when the tip was initially submitted by the user. |

| `updated\_at` | `timestamp with time zone` | `NOT NULL`, `DEFAULT now()` | Timestamp of when the tip record was last updated. Auto-updated by a database trigger. |

| `deleted\_at` | `timestamp with time zone` | `NULLABLE` | Timestamp for implementing soft deletion. Active tips have `deleted\_at IS NULL`. |

3\. PostgreSQL DDL

SQL

```

-- ENUM Type Definition for Moderation Status (if not already created globally)

DO $$ BEGIN

IF NOT EXISTS (SELECT 1 FROM pg\_type WHERE typname = 'content\_moderation\_status\_enum') THEN

CREATE TYPE public.content\_moderation\_status\_enum AS ENUM (

'pending\_approval',

'approved\_visible',

'rejected\_hidden',

'flagged\_for\_review\_by\_admin',

'archived\_by\_admin'

);

END IF;

END $$;

-- Table Definition for user\_waypoint\_short\_tips

CREATE TABLE public.user\_waypoint\_short\_tips (

id bigint NOT NULL GENERATED ALWAYS AS IDENTITY,

profile\_id uuid NOT NULL,

waypoint\_id integer NOT NULL, -- CRITICAL: Confirm this matches the data type of public.waypoints.id

tip\_text text NOT NULL,

language\_code text NOT NULL,

tip\_category\_code text NULL,

tip\_source\_platform text NULL,

moderation\_status public.content\_moderation\_status\_enum NOT NULL DEFAULT 'pending\_approval',

is\_publicly\_visible boolean NOT NULL GENERATED ALWAYS AS (moderation\_status = 'approved\_visible'::public.content\_moderation\_status\_enum AND deleted\_at IS NULL) STORED, -- Explicit cast for safety

is\_pinned\_by\_admin boolean NOT NULL DEFAULT false,

moderated\_by\_profile\_id uuid NULL,

moderation\_timestamp timestamptz NULL,

moderation\_notes\_internal text NULL,

created\_at timestamptz NOT NULL DEFAULT now(),

updated\_at timestamptz NOT NULL DEFAULT now(),

deleted\_at timestamptz NULL,

CONSTRAINT user\_waypoint\_short\_tips\_pkey PRIMARY KEY (id),

CONSTRAINT user\_waypoint\_short\_tips\_profile\_id\_fkey FOREIGN KEY (profile\_id) REFERENCES public.profiles(id) ON DELETE CASCADE,

CONSTRAINT user\_waypoint\_short\_tips\_waypoint\_id\_fkey FOREIGN KEY (waypoint\_id) REFERENCES public.waypoints(id) ON DELETE CASCADE,

CONSTRAINT user\_waypoint\_short\_tips\_language\_code\_fkey FOREIGN KEY (language\_code) REFERENCES public.languages\_master(language\_code) ON DELETE RESTRICT,

CONSTRAINT user\_waypoint\_short\_tips\_tip\_category\_code\_fkey FOREIGN KEY (tip\_category\_code) REFERENCES public.tip\_categories\_master(category\_code) ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT user\_waypoint\_short\_tips\_moderated\_by\_profile\_id\_fkey FOREIGN KEY (moderated\_by\_profile\_id) REFERENCES public.profiles(id) ON DELETE SET NULL,

CONSTRAINT user\_waypoint\_short\_tips\_tip\_text\_check CHECK ((char\_length(tip\_text) > 0 AND char\_length(tip\_text) <= 500))

);

-- Comments

COMMENT ON TABLE public.user\_waypoint\_short\_tips IS 'Stores brief, user-submitted textual tips (max 500 chars) about specific waypoints, subject to moderation. Version 2.1.1.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.id IS 'Unique identifier for the tip.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.profile\_id IS 'FK to profiles.id of the user who submitted the tip.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.waypoint\_id IS 'FK to waypoints.id of the waypoint this tip is about. CRITICAL: Confirm data type matches waypoints.id.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.tip\_text IS 'The content of the tip. Max 500 characters.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.language\_code IS 'Language code of the tip (e.g., ''en'', ''it''), FK to languages\_master.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.tip\_category\_code IS 'Optional category code for the tip from tip\_categories\_master. FK to tip\_categories\_master.category\_code.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.tip\_source\_platform IS 'Optional: Platform from which the tip was submitted (e.g., ''web\_app\_v1'').';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.moderation\_status IS 'The current moderation status of the tip. Uses public.content\_moderation\_status\_enum.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.is\_publicly\_visible IS 'Computed (STORED): True if moderation\_status is ''approved\_visible'' AND deleted\_at IS NULL. For client queries.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.is\_pinned\_by\_admin IS 'If true, administrators have marked this tip as particularly important.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.moderated\_by\_profile\_id IS 'FK to profiles.id of the admin/moderator who last took action.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.moderation\_timestamp IS 'Timestamp of the last moderation action.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.moderation\_notes\_internal IS 'Internal notes from the moderator, not typically public.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.created\_at IS 'Timestamp of when the tip was submitted by the user.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.updated\_at IS 'Timestamp of when the tip record was last updated. Auto-updated by trigger.';

COMMENT ON COLUMN public.user\_waypoint\_short\_tips.deleted\_at IS 'Timestamp for soft deletion, if a tip is retracted or removed by an admin.';

-- Indexes

CREATE INDEX IF NOT EXISTS idx\_user\_waypoint\_short\_tips\_visibility

ON public.user\_waypoint\_short\_tips (waypoint\_id, is\_publicly\_visible, is\_pinned\_by\_admin DESC, created\_at DESC);

CREATE INDEX IF NOT EXISTS idx\_user\_waypoint\_short\_tips\_moderation

ON public.user\_waypoint\_short\_tips (moderation\_status, created\_at ASC);

CREATE INDEX IF NOT EXISTS idx\_user\_waypoint\_short\_tips\_profile\_id

ON public.user\_waypoint\_short\_tips (profile\_id); -- Renamed for clarity

CREATE INDEX IF NOT EXISTS idx\_user\_waypoint\_short\_tips\_language\_code

ON public.user\_waypoint\_short\_tips (language\_code); -- Renamed for clarity

CREATE INDEX IF NOT EXISTS idx\_user\_waypoint\_short\_tips\_tip\_category\_code

ON public.user\_waypoint\_short\_tips (tip\_category\_code) WHERE tip\_category\_code IS NOT NULL; -- Renamed for clarity

-- Ensure the generic 'updated\_at' trigger function is available (defined elsewhere, e.g. with user\_waypoint\_votes)

-- CREATE OR REPLACE FUNCTION public.handle\_updated\_at() ...

-- Trigger for user\_waypoint\_short\_tips table 'updated\_at'

CREATE TRIGGER on\_user\_waypoint\_short\_tips\_updated\_at

BEFORE UPDATE ON public.user\_waypoint\_short\_tips

FOR EACH ROW

EXECUTE FUNCTION public.handle\_updated\_at();

-- RLS Policies (ensure public.check\_if\_user\_is\_moderator(UUID) function is defined)

ALTER TABLE public.user\_waypoint\_short\_tips ENABLE ROW LEVEL SECURITY;

CREATE POLICY "Pilgrims can insert their own tips"

ON public.user\_waypoint\_short\_tips FOR INSERT

TO authenticated

WITH CHECK (auth.uid() = profile\_id);

CREATE POLICY "Pilgrims can view publicly visible tips"

ON public.user\_waypoint\_short\_tips FOR SELECT

TO authenticated -- Also applicable to 'anon' role if API serves this view directly

USING (is\_publicly\_visible = true);

CREATE POLICY "Pilgrims can view their own tips"

ON public.user\_waypoint\_short\_tips FOR SELECT

TO authenticated

USING (auth.uid() = profile\_id);

CREATE POLICY "Pilgrims can update their own tips"

ON public.user\_waypoint\_short\_tips FOR UPDATE

TO authenticated

USING (auth.uid() = profile\_id)

WITH CHECK (

auth.uid() = profile\_id AND

(

-- Scenario 1: User is only changing deleted\_at (retracting/un-retracting an existing tip)

(OLD.deleted\_at IS DISTINCT FROM NEW.deleted\_at AND

OLD.tip\_text = NEW.tip\_text AND

OLD.language\_code = NEW.language\_code AND

OLD.tip\_category\_code IS NOT DISTINCT FROM NEW.tip\_category\_code AND

OLD.moderation\_status = NEW.moderation\_status -- Ensure moderation status is not changed by user

) OR

-- Scenario 2: Tip is still pending approval, user can edit content fields. Moderation status must remain pending.

(OLD.moderation\_status = 'pending\_approval'::public.content\_moderation\_status\_enum AND

NEW.moderation\_status = 'pending\_approval'::public.content\_moderation\_status\_enum)

)

);

-- Placeholder for moderator check function. Replace 'public.check\_if\_user\_is\_moderator' with actual implementation.

CREATE POLICY "Moderators can view all tips"

ON public.user\_waypoint\_short\_tips FOR SELECT

TO authenticated

USING (public.check\_if\_user\_is\_moderator(auth.uid())); -- Requires function definition

CREATE POLICY "Moderators can update moderation fields on tips"

ON public.user\_waypoint\_short\_tips FOR UPDATE

TO authenticated

USING (public.check\_if\_user\_is\_moderator(auth.uid())) -- Requires function definition

WITH CHECK (public.check\_if\_user\_is\_moderator(auth.uid())); -- Ensures only moderators can update, check applies to columns they can update

COMMENT ON POLICY "Pilgrims can insert their own tips" ON public.user\_waypoint\_short\_tips IS 'Allows authenticated users to submit new tips for waypoints.';

COMMENT ON POLICY "Pilgrims can view publicly visible tips" ON public.user\_waypoint\_short\_tips IS 'Allows authenticated users (and potentially anonymous via API) to view tips that are approved and not deleted.';

COMMENT ON POLICY "Pilgrims can view their own tips" ON public.user\_waypoint\_short\_tips IS 'Allows authenticated users to view all of their own submitted tips, regardless of moderation status.';

COMMENT ON POLICY "Pilgrims can update their own tips" ON public.user\_waypoint\_short\_tips IS 'Allows authenticated users to update their own tips, primarily for retraction (soft delete) or editing content if still pending approval. Business logic for edits on approved tips (e.g., resetting moderation status) is crucial.';

COMMENT ON POLICY "Moderators can view all tips" ON public.user\_waypoint\_short\_tips IS 'Allows users with moderator privileges to view all tips for moderation purposes.';

COMMENT ON POLICY "Moderators can update moderation fields on tips" ON public.user\_waypoint\_short\_tips IS 'Allows users with moderator privileges to update fields relevant to moderation (e.g., moderation\_status, is\_pinned\_by\_admin, moderation\_notes\_internal).';

```

4\. JSON Schema Mirror

JSON

```

{

"title": "user\_waypoint\_short\_tip",

"description": "Stores brief, user-submitted textual tips or mini-reviews (max 500 characters) about specific waypoints, subject to a moderation workflow. Version 2.1.1.",

"type": "object",

"properties": {

"id": {

"type": "integer",

"format": "int64",

"description": "Unique identifier for the tip.",

"readOnly": true

},

"profile\_id": {

"type": "string",

"format": "uuid",

"description": "FK to profiles.id of the user who submitted the tip."

},

"waypoint\_id": {

"type": "integer",

"description": "FK to waypoints.id of the waypoint this tip is about. CRITICAL: Confirm data type matches waypoints.id."

},

"tip\_text": {

"type": "string",

"maxLength": 500,

"description": "The content of the tip. Max length 500 enforced by DB constraint."

},

"language\_code": {

"type": "string",

"pattern": "^[a-z]{2}(-[A-Z]{2})?$",

"description": "Language code of the tip (e.g., 'en', 'it'), FK to languages\_master."

},

"tip\_category\_code": {

"type": ["string", "null"],

"description": "Optional category code for the tip (from tip\_categories\_master). FK to tip\_categories\_master.category\_code."

},

"tip\_source\_platform": {

"type": ["string", "null"],

"description": "Optional: Platform from which the tip was submitted (e.g., 'web\_app\_v1')."

},

"moderation\_status": {

"$ref": "#/definitions/content\_moderation\_status\_enum",

"default": "pending\_approval",

"description": "The current moderation status of the tip."

},

"is\_publicly\_visible": {

"type": "boolean",

"description": "Computed (STORED): True if moderation\_status is 'approved\_visible' AND deleted\_at IS NULL. ReadOnly.",

"readOnly": true

},

"is\_pinned\_by\_admin": {

"type": "boolean",

"default": false,

"description": "If true, administrators have marked this tip as particularly important for its waypoint."

},

"moderated\_by\_profile\_id": {

"type": ["string", "null"],

"format": "uuid",

"description": "FK to profiles.id of the admin/moderator who last took action."

},

"moderation\_timestamp": {

"type": ["string", "null"],

"format": "date-time",

"description": "Timestamp of when the last moderation action was taken."

},

"moderation\_notes\_internal": {

"type": ["string", "null"],

"description": "Internal notes from the moderator regarding their decision or edits."

},

"created\_at": {

"type": "string",

"format": "date-time",

"description": "Timestamp of when the tip was submitted by the user.",

"readOnly": true

},

"updated\_at": {

"type": "string",

"format": "date-time",

"description": "Timestamp of when the tip record was last updated. Auto-updated by trigger.",

"readOnly": true

},

"deleted\_at": {

"type": ["string", "null"],

"format": "date-time",

"description": "Timestamp for soft deletion, if a tip is retracted or removed by an admin.",

"readOnly": true

}

},

"required": [

"id",

"profile\_id",

"waypoint\_id",

"tip\_text",

"language\_code",

"moderation\_status",

"is\_publicly\_visible",

"is\_pinned\_by\_admin",

"created\_at",

"updated\_at"

],

"primary\_key": ["id"],

"definitions": {

"content\_moderation\_status\_enum": {

"type": "string",

"enum": [

"pending\_approval",

"approved\_visible",

"rejected\_hidden",

"flagged\_for\_review\_by\_admin",

"archived\_by\_admin"

]

}

}

}

```

5\. Relationships & Integrity

- Foreign Keys:

- `profile\_id` references `public.profiles(id)` (`ON DELETE CASCADE`).

- `waypoint\_id` references `public.waypoints(id)` (`ON DELETE CASCADE`).

- `language\_code` references `public.languages\_master(language\_code)` (`ON DELETE RESTRICT`).

- `tip\_category\_code` references `public.tip\_categories\_master(category\_code)` (`ON DELETE SET NULL ON UPDATE CASCADE`).

- `moderated\_by\_profile\_id` references `public.profiles(id)` (`ON DELETE SET NULL`).

- Generated Column: `is\_publicly\_visible` ensures data consistency for tip visibility.

- Mermaid ER Snippet:

Code snippet

```

erDiagram

profiles { uuid id PK }

waypoints { integer id PK "CRITICAL: Confirm type" }

languages\_master { text language\_code PK }

tip\_categories\_master { text category\_code PK }

user\_waypoint\_short\_tips {

bigint id PK

uuid profile\_id FK

integer waypoint\_id FK "CRITICAL: Confirm type"

text tip\_text "CHECK len<=500"

text language\_code FK

text tip\_category\_code FK

text tip\_source\_platform

content\_moderation\_status\_enum moderation\_status

boolean is\_publicly\_visible "GENERATED"

boolean is\_pinned\_by\_admin

uuid moderated\_by\_profile\_id FK

timestamptz moderation\_timestamp

text moderation\_notes\_internal

timestamptz created\_at

timestamptz updated\_at

timestamptz deleted\_at

}

profiles ||--o{ user\_waypoint\_short\_tips : "submits"

waypoints ||--o{ user\_waypoint\_short\_tips : "has\_tip\_about"

languages\_master ||--|| user\_waypoint\_short\_tips : "language\_of"

tip\_categories\_master ||--o{ user\_waypoint\_short\_tips : "categorized\_by"

profiles ||--o{ user\_waypoint\_short\_tips : "moderated\_by (admin)"

```

6\. Multilingual Strategy

- `tip\_text` is stored in the language it was submitted in, indicated by `language\_code` (FK to `languages\_master`). This allows for UI-level filtering or future UI-triggered translation features.

- `tip\_category\_code` links to `tip\_categories\_master` which handles multilingual category names via its own translatable fields (`default\_name`, `default\_description`) and the `public.translations` table.

7\. Role-Based Workflow & RLS Notes

- Key Fields for Workflow: `profile\_id`, `moderation\_status`, `is\_publicly\_visible`, `is\_pinned\_by\_admin`, `moderated\_by\_profile\_id`, `moderation\_timestamp`, `deleted\_at`.

- RLS Policies:

- Table RLS enabled.

- Authenticated users can insert their own tips.

- Authenticated users (and potentially anonymous via API) can view publicly visible tips.

- Authenticated users can view all their own tips.

- Authenticated users can update their own tips (primarily for retraction or editing PENDING tips; edits to approved tips need careful business rule definition).

- Moderators (via `public.check\_if\_user\_is\_moderator()`) can view all tips and update moderation-related fields.

8\. ENUM vs Lookup Discussion

- `tip\_category\_code` now correctly references `tip\_categories\_master` (a lookup table), which is excellent for i18n, icons, and admin management.

- `content\_moderation\_status\_enum` remains an ENUM, appropriate for its defined set of workflow states.

9\. UI/UX Enablement

- Columns Powering UI: `tip\_text`, `language\_code`, `tip\_category\_code` (with join to `tip\_categories\_master`), `is\_publicly\_visible`, `is\_pinned\_by\_admin`, `created\_at`, `profile\_id` (with join to `profiles`).

- Performance: Comprehensive indexing strategy (`idx\_user\_waypoint\_short\_tips\_visibility`, etc.) supports common query patterns. The `STORED` generated column `is\_publicly\_visible` also aids read performance.

10\. Auditing & Lifecycle Management

- Audit Columns: `created\_at`, `updated\_at`. `profile\_id` identifies the creator. `moderated\_by\_profile\_id` and `moderation\_timestamp` track moderation actions.

- Soft Deletion: `deleted\_at` for tip retractions.

- Lifecycle: `moderation\_status` and the generated `is\_publicly\_visible` manage the tip's visibility lifecycle.

11\. Scalability & Future-Proofing

- Partitioning: Unlikely needed for V1.

- Text Search (V2+): Plan for FTS on `tip\_text` is noted.

- Reporting/Flagging/Helpfulness (V2+): Schema allows for these extensions.

12\. Seed Data

- Not applicable; this table contains user-generated transactional data.

13\. Next-Action Checklist

1. 🔴 CRITICAL: Confirm `user\_waypoint\_short\_tips.waypoint\_id` Data Type: Rigorously verify and align the `integer` data type with `public.waypoints.id`. Adjust DDL if `waypoints.id` is different (e.g., `bigint`, `uuid`).

2. 🔴 CRITICAL: Define & Implement `public.check\_if\_user\_is\_moderator(UUID)`: This function is essential for moderator RLS policies to work correctly. Its logic needs to be defined based on how moderators are identified in the system.

3. 🔴 CRITICAL: Define and Implement User Tip Edit Policy: Finalize and rigorously test the RLS policy "Pilgrims can update their own tips" and any associated application logic or database triggers to ensure it correctly handles edits to tips in various states (pending, approved), especially regarding changes to `moderation\_status`.

4. 🟢 Verify PostgreSQL Version: Ensure PostgreSQL 12+ for the `STORED` generated column `is\_publicly\_visible` (typically fine on Supabase).

5. 🟢 `updated\_at` Trigger: Ensure `public.handle\_updated\_at()` and its trigger are correctly implemented and active.

6. 🟢 RLS Policies: Implement and test all other RLS policies.

7. 🟢 Application Logic: Develop backend logic for tip submission, moderation workflows, and displaying tips according to visibility rules and category joins.

8. 🟢 Documentation: Note the deliberate audit column strategy (using `profile\_id` as creator, `moderated\_by\_profile\_id` for moderation actions) and the reasoning.