**Summary of Proposed Plan:**

* **Step 1**: Update the 9 master table specification documents in Module 4a to add the recommended label (or name) index.
* **Step 2**: Update the 4.2 Accommodations.docx to add the GIN index on opening\_months and update its "Future Considerations".
* **Step 3**: Update the relevant junction table specifications' "Future Considerations" sections.
* **Step 4**: Create the 9 new, individual specification documents for the localized views (e.g., 4.3.v - v\_accommodation\_types\_localized.docx, etc.).

Does this sequence sound good to you? If so, we can start with updating the first master table specification document to add its label index.

[**https://gemini.google.com/u/1/app/fc0ce8702c9586a4**](https://gemini.google.com/u/1/app/fc0ce8702c9586a4)

**reviews Table (especially for Accommodations):**

* **Logical Placement:** This fits best as an extension of **Module 6.0 - User Interaction**, as it's primarily user-generated content related to waypoints. Alternatively, if reviews become very specific to accommodations with unique fields beyond general user feedback, it could be a new sub-table within **Module 4a.0 - Waypoint - Accommodations**, directly linking to accommodations.waypoint\_id.
* **Recommendation:** Start by conceptualizing it within **Module 6.0**, as it shares characteristics with user\_waypoint\_short\_tips (user authorship, moderation, linkage to waypoints and profiles). If it becomes highly specialized for accommodations, it could then be more tightly coupled with Module 4a.0.

**2.1. Dedicated reviews Table (especially for Accommodations):**

* **Action:** Since you'd like this, we need to add a new table specification to your documentation set.
* **Proposed Schema for public.accommodation\_reviews:** | Column Name | Data Type | Constraints | Description | | :--------------------------- | :--------------------- | :-------------------------------------------------------------------------------- | :--------------------------------------------------------------------------------------------------------------------------------------- | | id | BIGINT | PRIMARY KEY, GENERATED ALWAYS AS IDENTITY | Unique review identifier. | | accommodation\_waypoint\_id | BIGINT | NOT NULL, REFERENCES public.accommodations(waypoint\_id) ON DELETE CASCADE | Links to the specific accommodation being reviewed. | | profile\_id | UUID | NOT NULL, REFERENCES public.profiles(id) ON DELETE CASCADE | Author of the review. | | review\_title | TEXT | NULLABLE, CHECK (char\_length(review\_title) <= 255) | Optional title for the review. Translatable. | | review\_body | TEXT | NOT NULL, CHECK (char\_length(review\_body) > 0 AND char\_length(review\_body) <= 2000) | The main content of the review. Translatable. (Limit increased from "short tips"). | | language\_code | TEXT | NOT NULL, REFERENCES public.languages\_master(language\_code) ON DELETE RESTRICT | Language the review was originally written in. | | stay\_date | DATE | NULLABLE | Optional: when the pilgrim stayed at the accommodation. | | overall\_vote | vote\_type\_enum | NULLABLE | Optional: a simple thumbs up/down if you want to retain that alongside the text review. Could link to user\_waypoint\_votes or be standalone. | | moderation\_status | content\_moderation\_status\_enum | NOT NULL, DEFAULT 'pending\_approval' | Moderation status (reuse enum from Module 6.0). | | is\_publicly\_visible | BOOLEAN | NOT NULL, GENERATED ALWAYS AS (moderation\_status = 'approved\_visible' AND deleted\_at IS NULL) STORED | Auto-calculated visibility (requires PG12+). | | moderated\_by\_profile\_id | UUID | NULLABLE, REFERENCES public.profiles(id) ON DELETE SET NULL | Admin/moderator who actioned this review. | | moderation\_timestamp | TIMESTAMPTZ | NULLABLE | When the moderation action was taken. | | moderation\_notes\_internal | TEXT | NULLABLE | Internal notes for moderation. | | created\_at | TIMESTAMPTZ | NOT NULL, DEFAULT now() | | | updated\_at | TIMESTAMPTZ | NOT NULL, DEFAULT now() | | | deleted\_at | TIMESTAMPTZ | NULLABLE | For soft deletion by user or admin. |
* **Triggers:** Standard updated\_at trigger. Consider if inserting a review should also create/update a user\_waypoint\_votes record for consistency if overall\_vote is included.
* **RLS:** Similar to user\_waypoint\_short\_tips (user manages own, mods manage all, public reads approved).
* **Translations:** review\_title, review\_body link to public.translations.
* **Relevant Modules:** New table, closely related to Module 4a.0 (Accommodations) and Module 6.0 (User Interaction).

**3.1. Array Foreign Key Integrity (Reiteration from Phase 1 - focus on implementation now):**

* **Action:** For each table identified in Phase 1.1, write the specific DDL for its validation trigger function (similar to public.check\_event\_theme\_tags\_exist() from Module 4d.0) and CREATE TRIGGER statement. Document this in the "Triggers/Functions" section of that table's spec.
* **Relevant Modules:** 4.0, 4b.0, 4c.0, 4d.0.

**3.2. Deletion Strategy for Master Records Referenced in Arrays:**

* **Action:** For master tables like tags\_master, transport\_stop\_facilities\_master, visitor\_amenities\_master, etc.:
  1. Add is\_active BOOLEAN NOT NULL DEFAULT true if not present.
  2. Update the Array FK validation triggers (from step 3.1) on referencing tables to check is\_active = true in the master table for each ID in the array.
  3. Document that "retiring" a master entry is done by setting is\_active = false.
* **Relevant Modules:** 4.0, 4b.0, 4c.0, 4d.0.