<https://gemini.google.com/u/1/app/5b5f3092ace62463>

\* \* \* \* \*

### Updated Production-Ready Specification: `public.articles` Table

Production-Ready Specification: public.articles Table

Version: 1.0 (Checklist REV 05-18-25-A Applied)

Date: May 18, 2025

1\. Purpose & Primary Use-Cases

Mission Statement: To store and manage general content articles such as news, platform updates, or blog-style posts. It supports titles, slugs for URL generation, main body content, excerpts, author attribution, publication status, and optional associations with trails, regions, or towns, along with audit trails. Key translatable fields enable multilingual content delivery.

Key User-Story Touchpoints:

- Pilgrim: Reads published articles for news, updates, or information.

- Regional Manager/Admin: Creates, edits, and manages articles relevant to their scope or the entire platform.

- System/Developer: Queries articles for display, relies on `article\_status` for visibility, and `slug` for routing.

2\. Schema

| Column Name | Data Type | Constraints | Description |

| `id` | `BIGINT` | `PRIMARY KEY, GENERATED ALWAYS AS IDENTITY` | Unique article identifier. |

| `title` | `TEXT` | `NOT NULL, CHECK (char\_length(title) > 0 AND char\_length(title) <= 255)` | Article title. Max 255 chars. (Translatable via `public.translations`) |

| `slug` | `TEXT` | `NOT NULL, UNIQUE, CHECK (slug ~ '^[a-z0-9]+(?:-[a-z0-9]+)\*$' AND char\_length(slug) > 0 AND char\_length(slug) <= 255)` | URL-friendly slug, unique. Generated by application logic. Max 255 chars. |

| `body\_content` | `TEXT` | `NOT NULL` | Main content of the article (e.g., Markdown, HTML). (Translatable via `public.translations`) |

| `excerpt` | `TEXT` | `CHECK (excerpt IS NULL OR (char\_length(excerpt) > 0 AND char\_length(excerpt) <= 500))` | Short summary or excerpt of the article. Max 500 chars. (Translatable via `public.translations`) |

| `author\_profile\_id` | `UUID` | `REFERENCES public.profiles(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Profile ID of the author (likely a regional manager or admin). |

| `publication\_date` | `TIMESTAMPTZ` | | Timestamp when the article is/was published. Can be set to a future date for scheduled publishing. |

| `article\_status` | `public.content\_visibility\_status\_enum` | `NOT NULL, DEFAULT 'draft'::public.content\_visibility\_status\_enum` | Lifecycle status (e.g., 'draft', 'published', 'archived'). Default is 'draft'. |

| `featured\_image\_media\_id` | `UUID` | `REFERENCES public.media(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Optional featured image for the article. |

| `associated\_trail\_id` | `BIGINT` | `REFERENCES public.trails(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Optional: Link to a specific trail. |

| `associated\_region\_id` | `BIGINT` | `REFERENCES public.regions(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Optional: Link to a specific region. |

| `associated\_town\_id` | `BIGINT` | `REFERENCES public.towns(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Optional: Link to a specific town. |

| `tags` | `TEXT[]` | | Keywords for categorization or searching articles. |

| `created\_at` | `TIMESTAMPTZ` | `NOT NULL, DEFAULT now()` | Timestamp of article record creation. |

| `updated\_at` | `TIMESTAMPTZ` | `NOT NULL, DEFAULT now()` | Timestamp of last article record update (auto-updated by trigger). |

| `created\_by\_profile\_id` | `UUID` | `REFERENCES public.profiles(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Profile ID of the user who created the article record. |

| `updated\_by\_profile\_id` | `UUID` | `REFERENCES public.profiles(id) ON DELETE SET NULL ON UPDATE NO ACTION` | Profile ID of the user who last updated the article record. |

| `deleted\_at` | `TIMESTAMPTZ` | | Timestamp for soft deletion of the article. |

3\. PostgreSQL DDL

SQL

```

-- Ensure content\_visibility\_status\_enum is defined globally or as part of a prerequisite script.

-- Example:

-- CREATE TYPE public.content\_visibility\_status\_enum AS ENUM ('draft', 'pending\_review', 'published', 'archived', 'private\_link');

CREATE TABLE public.articles (

id BIGINT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,

title TEXT NOT NULL,

slug TEXT NOT NULL UNIQUE,

body\_content TEXT NOT NULL,

excerpt TEXT NULL,

author\_profile\_id UUID NULL REFERENCES public.profiles(id) ON DELETE SET NULL ON UPDATE NO ACTION,

publication\_date TIMESTAMPTZ NULL,

article\_status public.content\_visibility\_status\_enum NOT NULL DEFAULT 'draft'::public.content\_visibility\_status\_enum,

featured\_image\_media\_id UUID NULL REFERENCES public.media(id) ON DELETE SET NULL ON UPDATE NO ACTION,

associated\_trail\_id BIGINT NULL REFERENCES public.trails(id) ON DELETE SET NULL ON UPDATE NO ACTION,

associated\_region\_id BIGINT NULL REFERENCES public.regions(id) ON DELETE SET NULL ON UPDATE NO ACTION,

associated\_town\_id BIGINT NULL REFERENCES public.towns(id) ON DELETE SET NULL ON UPDATE NO ACTION,

tags TEXT[] NULL,

created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),

updated\_at TIMESTAMPTZ NOT NULL DEFAULT now(),

created\_by\_profile\_id UUID NULL REFERENCES public.profiles(id) ON DELETE SET NULL ON UPDATE NO ACTION,

updated\_by\_profile\_id UUID NULL REFERENCES public.profiles(id) ON DELETE SET NULL ON UPDATE NO ACTION,

deleted\_at TIMESTAMPTZ NULL,

CONSTRAINT check\_article\_title\_length CHECK (char\_length(title) > 0 AND char\_length(title) <= 255),

CONSTRAINT check\_article\_slug\_format CHECK (slug ~ '^[a-z0-9]+(?:-[a-z0-9]+)\*$' AND char\_length(slug) > 0 AND char\_length(slug) <= 255),

CONSTRAINT check\_article\_excerpt\_length CHECK (excerpt IS NULL OR (char\_length(excerpt) > 0 AND char\_length(excerpt) <= 500))

);

-- Indexes

CREATE INDEX idx\_articles\_slug ON public.articles (slug); -- Unique constraint already creates an index, but explicit naming can be preferred.

CREATE INDEX idx\_articles\_author\_profile\_id ON public.articles (author\_profile\_id) WHERE author\_profile\_id IS NOT NULL;

CREATE INDEX idx\_articles\_publication\_date ON public.articles (publication\_date DESC NULLS LAST) WHERE publication\_date IS NOT NULL;

CREATE INDEX idx\_articles\_status ON public.articles (article\_status);

CREATE INDEX idx\_articles\_featured\_image\_media\_id ON public.articles (featured\_image\_media\_id) WHERE featured\_image\_media\_id IS NOT NULL;

CREATE INDEX idx\_articles\_associated\_trail\_id ON public.articles (associated\_trail\_id) WHERE associated\_trail\_id IS NOT NULL;

CREATE INDEX idx\_articles\_associated\_region\_id ON public.articles (associated\_region\_id) WHERE associated\_region\_id IS NOT NULL;

CREATE INDEX idx\_articles\_associated\_town\_id ON public.articles (associated\_town\_id) WHERE associated\_town\_id IS NOT NULL;

CREATE INDEX idx\_articles\_tags ON public.articles USING GIN (tags) WHERE tags IS NOT NULL;

CREATE INDEX idx\_articles\_created\_by\_profile\_id ON public.articles (created\_by\_profile\_id) WHERE created\_by\_profile\_id IS NOT NULL;

CREATE INDEX idx\_articles\_updated\_by\_profile\_id ON public.articles (updated\_by\_profile\_id) WHERE updated\_by\_profile\_id IS NOT NULL;

CREATE INDEX idx\_articles\_deleted\_at ON public.articles (deleted\_at) WHERE deleted\_at IS NOT NULL;

-- Comments

COMMENT ON TABLE public.articles IS 'Stores general content articles like news, platform updates, or blog-style posts. Module 8. Version 1.0.';

COMMENT ON COLUMN public.articles.id IS 'PK. Unique article identifier. [cite: 3]';

COMMENT ON COLUMN public.articles.title IS 'Article title. Max 255 chars. (Translatable via public.translations) [cite: 4]';

COMMENT ON COLUMN public.articles.slug IS 'URL-friendly slug, unique, lowercase, alphanumeric with hyphens. Max 255 chars. Generated by application logic. [cite: 5, 6]';

COMMENT ON COLUMN public.articles.body\_content IS 'Main content of the article (e.g., Markdown, HTML). (Translatable via public.translations) [cite: 7]';

COMMENT ON COLUMN public.articles.excerpt IS 'Short summary or excerpt. Max 500 chars. (Translatable via public.translations) [cite: 8]';

COMMENT ON COLUMN public.articles.author\_profile\_id IS 'FK to profiles.id. Author of the article. [cite: 9]';

COMMENT ON COLUMN public.articles.publication\_date IS 'Timestamp when the article is/was published. Nullable for drafts or if not yet determined. [cite: 10]';

COMMENT ON COLUMN public.articles.article\_status IS 'Lifecycle status using content\_visibility\_status\_enum (e.g., draft, published, archived). Default is draft. [cite: 11]';

COMMENT ON COLUMN public.articles.featured\_image\_media\_id IS 'FK to media.id. Optional featured image for the article. [cite: 12, 13]';

COMMENT ON COLUMN public.articles.associated\_trail\_id IS 'FK to trails.id. Optional link to a specific trail. [cite: 14]';

COMMENT ON COLUMN public.articles.associated\_region\_id IS 'FK to regions.id. Optional link to a specific region. [cite: 15, 16]';

COMMENT ON COLUMN public.articles.associated\_town\_id IS 'FK to towns.id. Optional link to a specific town. [cite: 17, 18]';

COMMENT ON COLUMN public.articles.tags IS 'Array of keywords for categorization or searching articles. [cite: 19]';

COMMENT ON COLUMN public.articles.created\_at IS 'Timestamp of article record creation. [cite: 20]';

COMMENT ON COLUMN public.articles.updated\_at IS 'Timestamp of last article record update (auto-updated by trigger). [cite: 20]';

COMMENT ON COLUMN public.articles.created\_by\_profile\_id IS 'Profile ID of the user who created the article record. FK to profiles.id. [cite: 21]';

COMMENT ON COLUMN public.articles.updated\_by\_profile\_id IS 'Profile ID of the user who last updated the article record. FK to profiles.id. [cite: 22]';

COMMENT ON COLUMN public.articles.deleted\_at IS 'Timestamp for soft deletion. Filters should exclude records where deleted\_at IS NOT NULL. [cite: 23]';

```

4\. Triggers/Functions

SQL

```

-- Trigger for general updated\_at timestamp

CREATE TRIGGER handle\_articles\_updated\_at

BEFORE UPDATE ON public.articles

FOR EACH ROW

EXECUTE FUNCTION extensions.moddatetime('updated\_at');

COMMENT ON TRIGGER handle\_articles\_updated\_at ON public.articles IS 'Automatically updates the updated\_at timestamp on row modification.';

-- Trigger function to clean up related translations on delete

CREATE OR REPLACE FUNCTION public.cleanup\_articles\_translations()

RETURNS TRIGGER AS $$

BEGIN

DELETE FROM public.translations

WHERE table\_identifier = 'articles'

AND column\_identifier IN ('title', 'body\_content', 'excerpt') -- Specify translatable columns

AND row\_foreign\_key = OLD.id::text;

RETURN OLD;

END;

$$ LANGUAGE plpgsql SECURITY DEFINER;

COMMENT ON FUNCTION public.cleanup\_articles\_translations() IS 'Removes related entries from public.translations for title, body\_content, and excerpt when an article is deleted. Runs as SECURITY DEFINER. Ensure search\_path is secure if function complexity increases.';

-- Apply the cleanup trigger for translations

CREATE TRIGGER trigger\_cleanup\_articles\_translations\_after\_delete

AFTER DELETE ON public.articles

FOR EACH ROW

EXECUTE FUNCTION public.cleanup\_articles\_translations();

COMMENT ON TRIGGER trigger\_cleanup\_articles\_translations\_after\_delete ON public.articles IS 'After an article is deleted, removes its associated translations from the public.translations table.';

```

5\. JSON Schema Mirror

JSON

```

{

"title": "article",

"description": "Stores general content articles like news, platform updates, or blog-style posts. Module 8. Version 1.0.",

"type": "object",

"properties": {

"id": {

"type": "integer",

"format": "int64",

"description": "PK. Unique article identifier. [cite: 3]",

"readOnly": true

},

"title": {

"type": "string",

"description": "Article title. Max 255 chars. (Translatable via public.translations) [cite: 4]",

"maxLength": 255

},

"slug": {

"type": "string",

"description": "URL-friendly slug, unique, lowercase, alphanumeric with hyphens. Max 255 chars. Generated by application logic. [cite: 5, 6]",

"pattern": "^[a-z0-9]+(?:-[a-z0-9]+)\*$",

"maxLength": 255

},

"body\_content": {

"type": "string",

"description": "Main content of the article (e.g., Markdown, HTML). (Translatable via public.translations) [cite: 7]"

},

"excerpt": {

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

"description": "Short summary or excerpt. Max 500 chars. (Translatable via public.translations) [cite: 8]",

"maxLength": 500

},

"author\_profile\_id": {

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

"format": "uuid",

"description": "FK to profiles.id. Author of the article. [cite: 9]"

},

"publication\_date": {

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

"format": "date-time",

"description": "Timestamp when the article is/was published. Nullable for drafts or if not yet determined. [cite: 10]"

},

"article\_status": {

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

"default": "draft",

"description": "Lifecycle status using content\_visibility\_status\_enum (e.g., draft, published, archived). [cite: 11]"

},

"featured\_image\_media\_id": {

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

"format": "uuid",

"description": "FK to media.id. Optional featured image for the article. [cite: 12, 13]"

},

"associated\_trail\_id": {

"type": ["integer", "null"],

"format": "int64",

"description": "FK to trails.id. Optional link to a specific trail. [cite: 14]"

},

"associated\_region\_id": {

"type": ["integer", "null"],

"format": "int64",

"description": "FK to regions.id. Optional link to a specific region. [cite: 15, 16]"

},

"associated\_town\_id": {

"type": ["integer", "null"],

"format": "int64",

"description": "FK to towns.id. Optional link to a specific town. [cite: 17, 18]"

},

"tags": {

"type": ["array", "null"],

"items": { "type": "string" },

"description": "Array of keywords for categorization or searching articles. [cite: 19]"

},

"created\_at": {

"type": "string",

"format": "date-time",

"description": "Timestamp of article record creation. [cite: 20]",

"readOnly": true

},

"updated\_at": {

"type": "string",

"format": "date-time",

"description": "Timestamp of last article record update (auto-updated by trigger). [cite: 20]",

"readOnly": true

},

"created\_by\_profile\_id": {

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

"format": "uuid",

"description": "Profile ID of the user who created the article record. FK to profiles.id. [cite: 21]"

},

"updated\_by\_profile\_id": {

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

"format": "uuid",

"description": "Profile ID of the user who last updated the article record. FK to profiles.id. [cite: 22]"

},

"deleted\_at": {

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

"format": "date-time",

"description": "Timestamp for soft deletion. Filters should exclude records where deleted\_at IS NOT NULL. [cite: 23]",

"readOnly": true

}

},

"required": [

"id",

"title",

"slug",

"body\_content",

"article\_status",

"created\_at",

"updated\_at"

],

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

"unique\_constraints": [

{"columns": ["slug"], "name": "articles\_slug\_key"}

],

"foreign\_keys": [

{"columns": ["author\_profile\_id"], "references\_table": "public.profiles", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"},

{"columns": ["featured\_image\_media\_id"], "references\_table": "public.media", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"},

{"columns": ["associated\_trail\_id"], "references\_table": "public.trails", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"},

{"columns": ["associated\_region\_id"], "references\_table": "public.regions", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"},

{"columns": ["associated\_town\_id"], "references\_table": "public.towns", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"},

{"columns": ["created\_by\_profile\_id"], "references\_table": "public.profiles", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"},

{"columns": ["updated\_by\_profile\_id"], "references\_table": "public.profiles", "references\_columns": ["id"], "on\_delete": "SET NULL", "on\_update": "NO ACTION"}

],

"definitions": {

"content\_visibility\_status\_enum": {

"type": "string",

"description": "Describes the visibility status of content (actual enum values defined globally). [cite: 11]",

"enum": ["draft", "pending\_review", "published", "archived", "private\_link"]

}

}

}

```

6\. Relationships & Integrity

- Primary Key: `id` (BIGINT).

- Foreign Keys:

- `author\_profile\_id` references `public.profiles(id)`.

- `featured\_image\_media\_id` references `public.media(id)`.

- `associated\_trail\_id` references `public.trails(id)`.

- `associated\_region\_id` references `public.regions(id)`.

- `associated\_town\_id` references `public.towns(id)`.

- `created\_by\_profile\_id` references `public.profiles(id)`.

- `updated\_by\_profile\_id` references `public.profiles(id)`.

- All FKs use `ON DELETE SET NULL` and `ON UPDATE NO ACTION`.

- Slug Uniqueness: Enforced by `UNIQUE` constraint on `slug` column.

- Mermaid ER Snippet:

Code snippet

```

erDiagram

profiles ||--o{ articles : "authored\_by"

media ||--o{ articles : "featured\_in"

trails ||--o{ articles : "associated\_with\_trail"

regions ||--o{ articles : "associated\_with\_region"

towns ||--o{ articles : "associated\_with\_town"

translations }o--|| articles : "translates (conceptually)"

articles {

BIGINT id PK

TEXT title "Translatable"

TEXT slug UK

TEXT body\_content "Translatable"

TEXT excerpt "Translatable"

UUID author\_profile\_id FK

TIMESTAMPTZ publication\_date

content\_visibility\_status\_enum article\_status

UUID featured\_image\_media\_id FK

BIGINT associated\_trail\_id FK

BIGINT associated\_region\_id FK

BIGINT associated\_town\_id FK

TEXT[] tags

TIMESTAMPTZ created\_at

TIMESTAMPTZ updated\_at

UUID created\_by\_profile\_id FK

UUID updated\_by\_profile\_id FK

TIMESTAMPTZ deleted\_at

}

```

7\. Multilingual Strategy

- Translatable Fields: `title`, `body\_content`, `excerpt` store their primary reference language text directly in this table. Translated versions are stored in `public.translations` with `table\_identifier = 'articles'`, `column\_identifier` matching the field name, and `row\_foreign\_key = articles.id::text`.

- Orphan Cleanup: An `AFTER DELETE` trigger on `public.articles` (`trigger\_cleanup\_articles\_translations\_after\_delete`) calls `public.cleanup\_articles\_translations()` to remove related entries from `public.translations`.

8\. Role-Based Workflow & RLS Notes

- Key Fields for Roles/Workflows: `author\_profile\_id`, `article\_status`, `publication\_date`, `deleted\_at`.

- Recommended Row-Level-Security (RLS) Policies:

- Enable RLS: `ALTER TABLE public.articles ENABLE ROW LEVEL SECURITY;`

- Force RLS for table owners: `ALTER TABLE public.articles FORCE ROW LEVEL SECURITY;`

- 🟢 Policy 1: Public users can read published, non-deleted articles.

SQL

```

CREATE POLICY "Public can read published articles"

ON public.articles FOR SELECT

USING (article\_status = 'published'::public.content\_visibility\_status\_enum AND deleted\_at IS NULL);

```

- 🟢 Policy 2: Authors can manage their own articles if they are in 'draft' or 'pending\_review' status and not deleted.

SQL

```

CREATE POLICY "Authors can manage their own draft or pending articles"

ON public.articles FOR ALL

USING (author\_profile\_id = auth.uid() AND article\_status IN ('draft'::public.content\_visibility\_status\_enum, 'pending\_review'::public.content\_visibility\_status\_enum) AND deleted\_at IS NULL)

WITH CHECK (author\_profile\_id = auth.uid() AND article\_status IN ('draft'::public.content\_visibility\_status\_enum, 'pending\_review'::public.content\_visibility\_status\_enum));

```

- 🟠 Policy 3: Users with 'admin\_platform' or 'regional\_content\_manager' roles can manage all articles. \*(Note: For 'regional\_content\_manager', a more granular check involving `associated\_region\_id`, `associated\_trail\_id`, or `associated\_town\_id` against the manager's assigned areas would be ideal via a helper function. This is a simplified version for now.)\*

SQL

```

CREATE POLICY "Admins and Regional Managers can manage all articles"

ON public.articles FOR ALL

USING (

(public.has\_role('admin\_platform') OR public.has\_role('admin\_super'))

OR

(public.has\_role('regional\_content\_manager')) -- Add additional checks for regional managers if needed: AND public.is\_manager\_for\_article\_context(id)

)

WITH CHECK (

(public.has\_role('admin\_platform') OR public.has\_role('admin\_super'))

OR

(public.has\_role('regional\_content\_manager')) -- Add additional checks for regional managers if needed

);

```

9\. ENUM vs Lookup Discussion

- 🟢 `article\_status` uses the existing `public.content\_visibility\_status\_enum`, which is appropriate for managing content lifecycle states.

10\. UI/UX Enablement

- Columns powering UI: `title` (translated), `slug` (for URLs), `body\_content` (translated), `excerpt` (translated), `author\_profile\_id` (for author display), `publication\_date`, `featured\_image\_media\_id`, `tags`.

- Indexed Fields for UX Performance: PK, `slug`, `author\_profile\_id`, `publication\_date`, `article\_status`, `tags` (GIN). FKs are indexed.

11\. Auditing & Lifecycle Management

- Audit Columns: `created\_at`, `updated\_at` (auto-managed via `extensions.moddatetime` trigger). `created\_by\_profile\_id` and `updated\_by\_profile\_id` for tracking user actions.

- Lifecycle: Managed by `article\_status` enum and `publication\_date`.

- Soft Deletes: Implemented via `deleted\_at` timestamp. Application logic should filter out soft-deleted records.

12\. Scalability & Future-Proofing

- Scalability: The table structure is standard. For very high volumes, `publication\_date` or `article\_status` could be candidates for future partitioning.

- Future-Proofing:

- `tags TEXT[]` allows flexible categorization.

- Consider Full-Text Search (FTS) on `title`, `body\_content`, `excerpt`, and `tags` for V2+ to enhance search capabilities.

- The associated `public.article\_media` table (specified separately) provides support for multiple images/media per article.

13\. Seed Data

- ⚪ N/A - Article records are typically dynamic content and not seeded, unless specific template or system articles are required from launch.

14\. Next-Action Checklist

- 🔴 Dependency Check: Ensure `public.content\_visibility\_status\_enum` (used by `article\_status`) exists and contains appropriate values (e.g., 'draft', 'pending\_review', 'published', 'archived'). This table also depends on `public.profiles`, `public.media`, `public.trails`, `public.regions`, and `public.towns` for its foreign keys. The translation cleanup trigger depends on `public.translations`.

- 🔴 Create Table: Execute DDL for `public.articles`.

- 🔴 Create Indexes: Execute DDL for all defined indexes.

- 🔴 Implement Triggers:

- Apply `handle\_articles\_updated\_at` trigger (using `extensions.moddatetime`).

- Create and apply `public.cleanup\_articles\_translations()` function and `trigger\_cleanup\_articles\_translations\_after\_delete` trigger.

- 🔴 Implement RLS Policies: Define and apply RLS policies as outlined. Refine 'regional\_content\_manager' policy with appropriate helper functions if complex geographical restrictions are needed.

- 🟠 Application Logic:

- Develop logic for `slug` generation (e.g., from title, ensuring uniqueness).

- Ensure application correctly populates `author\_profile\_id`, `created\_by\_profile\_id`, and `updated\_by\_profile\_id`.

- Handle the display of translated content by querying `public.translations`.

- 🟢 Future: Plan for Full-Text Search implementation on relevant text fields.

- 🟢 Related Tables: Ensure `public.article\_media` and `public.media\_roles\_master` are also implemented to support full multimedia capabilities as discussed.