

Estructura de proyecto (apps)

Proyecto Django (config):

- `config/` (settings, urls, wsgi, asgi)

Apps:

1. `accounts`

- Login/registro/logout (usando `User` por defecto de Django)
- (Opcional dentro del mismo alcance) vistas/urls para autenticación

2. `profiles`

- Perfil público y edición de perfil
- Foto de perfil y datos básicos extendidos

3. `articles`

- Artículos del desarrollador (contenido HTML)
- Comentarios
- Reacciones like/dislike

Tablas (modelos) y nombres

1) Usuario (Django por defecto)

- **Tabla:** `auth_user`
- **Modelo:** `django.contrib.auth.models.User`

2) Profiles

App: profiles

- **Tabla:** profiles_profile
- **Modelo:** Profile
- **Campos típicos:**
 - id
 - user_id (OneToOne con auth_user)
 - display_name (o nombre público)
 - bio (texto corto)
 - avatar (ImageField / path)
 - created_at, updated_at

Relación:

- auth_user (1) ↔ (1) profiles_profile
-

3) Articles

App: articles

A) Artículos

- **Tabla:** articles_article
- **Modelo:** Article
- **Campos típicos:**
 - id
 - title

- `slug`
- `summary` (opcional)
- `content_html` (HTML completo del artículo)
- `status` (draft/published)
- `published_at`
- `created_at, updated_at`
- `author_id` (FK a `auth_user`, pero en la práctica será el dev/admin)

Relación:

- `auth_user (1) → (N) articles_article`

Nota clave: `content_html` es el campo que guarda el HTML para títulos, párrafos, bloques de código, spans con color, etc.

B) Comentarios

- **Tabla:** `articles_comment`
- **Modelo:** `Comment`
- **Campos típicos:**
 - `id`
 - `article_id` (FK a `articles_article`)
 - `user_id` (FK a `auth_user`)
 - `body` (texto del comentario)
 - `created_at, updated_at`
 - `is_active` (moderación simple)

Relaciones:

- `articles_article` (1) → (N) `articles_comment`
- `auth_user` (1) → (N) `articles_comment`

C) Reacciones (Like / Dislike)

- **Tabla:** `articles_reaction`
- **Modelo:** `Reaction`
- **Campos típicos:**
 - `id`
 - `article_id` (FK a `articles_article`)
 - `user_id` (FK a `auth_user`)
 - `value` (ej: `LIKE / DISLIKE` o `1 / -1`)
 - `created_at`, `updated_at`

Restricción lógica (a nivel de modelo/DB):

- **Un usuario solo puede tener 1 reacción por artículo**
(unique: `user_id + article_id`)

Relaciones:

- `articles_article` (1) → (N) `articles_reaction`
- `auth_user` (1) → (N) `articles_reaction`

Resumen tipo “mapa”

- `auth_user`
 - 1–1 `profiles_profile`

- 1-N articles_article
- 1-N articles_comment
- 1-N articles_reaction
- articles_article
 - 1-N articles_comment
 - 1-N articles_reaction