Backend Basics: Save "Offer Help" Messages

Goal:
- Save helper messages when a user clicks "Ofrecer ayuda" and submits.
- Return each helper user's own sent messages.

1) Create table in Supabase

Run this SQL:

```sql
create extension if not exists "uuid-ossp";

create table if not exists public.offer_help_messages (
  id uuid primary key default uuid_generate_v4(),
  helper_user_id uuid not null references public.app_users(id),
  need_help_post_id text not null,
  target_post_title text,
  target_category text,
  helper_name text not null,
  helper_contact text not null,
  message text not null,
  created_at timestamptz not null default now()
);
```

Also add this table to `backend/src/main/resources/db/create_tables.sql`.

2) Add backend files (same project pattern)

Create:
- `entity/OfferHelpMessage.java`
- `repository/OfferHelpMessageRepository.java`
- `service/OfferHelpMessageService.java`
- `controller/OfferHelpMessageController.java`
- DTOs for request/response

Repository should include:
- `findByHelperUserIdOrderByCreatedAtDesc(UUID helperUserId)`

3) Add 2 endpoints

- `POST /api/offer-help/messages` (authenticated)
- `GET /api/offer-help/messages/mine` (authenticated)

In controller:
- Read logged-in user from `AuthPrincipal` (same style as `InviteController`).
- Use `helperUserId` from auth, not from request body.

4) Expected payload and response

Request example:

```json
{
  "targetPostId": "p1",
  "targetPostTitle": "Necesito apoyo con alimentos esta semana",
  "targetCategory": "Alimentos",
  "helperName": "Maria",
  "helperContact": "801-555-0100",
  "message": "Puedo ayudarte con una despensa y transporte."
}
```

Create response example:

```json
{
  "id": "uuid",
  "targetPostId": "p1",
  "targetPostTitle": "Necesito apoyo con alimentos esta semana",
  "targetCategory": "Alimentos",
  "helperName": "Maria",
  "helperContact": "801-555-0100",
  "message": "Puedo ayudarte con una despensa y transporte.",
  "createdAt": "2026-02-14T12:00:00.000Z"
}
```

Error format:

```json
{ "error": "Unauthorized" }
```

5) Frontend switch when backend is ready

Set:

```bash
NEXT_PUBLIC_OFFERHELP_MODE=api
```

Frontend is already wired and will call:
- `POST /api/offer-help/messages`
- `GET /api/offer-help/messages/mine`