

# Build a simple Twitter clone using Next.js, Supabase, and Redux

## Goal

Create a basic Twitter style web app where users can sign up, log in, post tweets, and see a feed of tweets from all users.

## Tech stack

- Frontend: Next.js (App Router if you are comfortable)
- State management: Redux Toolkit
- Backend as a service: Supabase (Auth + Database + API)
- Styling: Tailwind CSS (or simple CSS if you prefer, but Tailwind is a bonus)

## Core features to implement

### 1. Authentication with Supabase

- Email + password signup and login
- Supabase Auth for session handling
- Persist user session in the app (user stays logged in on refresh)

### 2. Database using Supabase

Create tables at minimum:

- `profiles`
  - `id` (uuid, primary key, same as auth user id)
  - `username` (text, unique)
  - `full_name` (text, optional)
  - `avatar_url` (text, optional)

- **tweets**
  - id** (uuid, primary key)
  - user\_id** (uuid, foreign key to profiles / auth users)
  - content** (text, max 280 characters)
  - created\_at** (timestamp)

### 3. Tweet feed (home page)

- Show a list of recent tweets sorted by **created\_at** descending
- Each tweet shows:
  - Username
  - Content
  - Created time (for example "5 min ago")
- Only show the tweet box if user is logged in

### 4. Create tweet

- At the top of the feed show a textbox
- User can type up to 280 characters
- On submit:
  - Save tweet to Supabase
  - Update feed without full page refresh

### 5. User profile page

- Route like **/u/[username]**
- Show user info (username, full name, avatar if set)
- Show list of only that user's tweets

## 6. Redux integration

Use Redux Toolkit to manage client side state. At minimum:

- `authSlice`
  - Stores current user session info (id, email, username)
  - Actions: setUser, clearUser
- `tweetsSlice`
  - Stores list of tweets currently loaded
  - Actions: setTweets, addTweet, setLoading, setError

## 7. Use Redux for:

- Storing user session data after login
- Storing tweets list for the feed

## Pages to build

- `/login`
  - Login form, link to signup
- `/signup`
  - Signup form, ask for username and email + password
- `/` (Home)
  - If not logged in, redirect to login
  - If logged in, show:
    - Tweet input box
    - List of tweets
- `/u/[username]`
  - Profile page with that user's tweets

## **UI requirements**

- Clean and minimal layout similar to Twitter:
  - Left side simple header or top nav bar
  - Center column with tweets
- Mobile responsive
- Basic error and loading states
  - Show loading indicator when fetching tweets
  - Show a simple error message if something fails

## **What you must deliver**

1. Next.js project in a Git repository
2. README with:
  - Setup steps (how to run locally)
  - Required environment variables for Supabase
  - Any notes or limitations
3. Deployed version (bonus, if possible)
  - For example on Vercel, connected to Supabase project

## **Acceptance criteria**

- User can sign up and log in using Supabase
- Logged in user can create a tweet and it appears instantly in the feed
- Feed shows tweets from all users, sorted by newest first
- Redux is correctly used for auth and tweet state
- Code is reasonably clean and structured (components, slices, services)
- App works on desktop and mobile sized screens