

# Sprint #2 Kickoff & Authentication Setup

---

## ADR-001: Tech Stack Selection

**Date:** 19 September 2025

**Status:** Accepted

**Version:** 1.0

**Author:** Team CPE-5 (Dangerless Project)

## Context

The Dangerless application aims to provide travelers with real-time neighborhood safety ratings and local safety news alerts. The system requires:

- High-performance rendering of interactive maps and geospatial overlays.
- Real-time updates for safety events and community news.
- Secure authentication and user management.
- A fast, type-safe, maintainable development workflow suitable for an MVP.

The chosen stack must minimize infrastructure complexity while remaining scalable and avoiding vendor lock-in.

## Decision Drivers

- Real-time geospatial data visualization and querying.
- Low operational overhead and managed hosting.
- Strong developer productivity through TypeScript and modern frameworks.
- Rapid prototyping and MVP delivery within academic constraints.
- Cross-browser and cross-device compatibility.
- Secure, standards-based authentication (OAuth 2.0).

## Decision

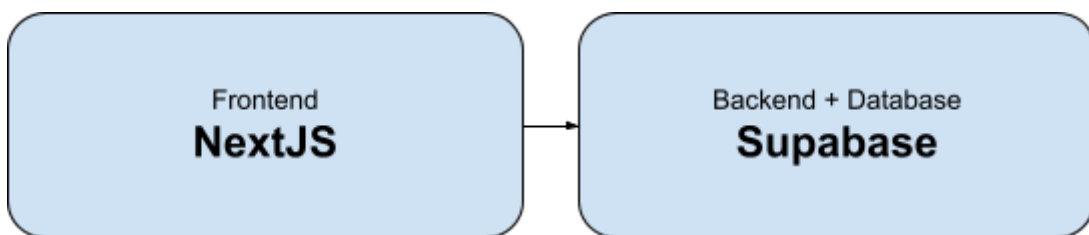
- **Frontend**
  - Framework: Next.js 14 (App Router) + TypeScript
  - Styling: Tailwind CSS
  - Maps: MapLibre GL JS (open, no vendor lock; supports vector tiles, polygon overlays, clustering)
  - Geocoding: Nominatim (for forward and reverse geocoding)
  - Data & State: TanStack Query (cache & sync with realtime invalidation)
  - **Rationale:** SSR/ISR for fast first paint, RSC for data isolation, strong TS support

- **Backend & Database**

- Platform: Supabase
- Database: Postgres + PostGIS
- Auth: Supabase Auth (Google OAuth 2.0, PKCE)
- Realtime: Supabase Realtime (WS)
- **Rationale:** Tight integration, managed ops, row-level security (RLS) enforces least privilege

- **Hosting/CI**

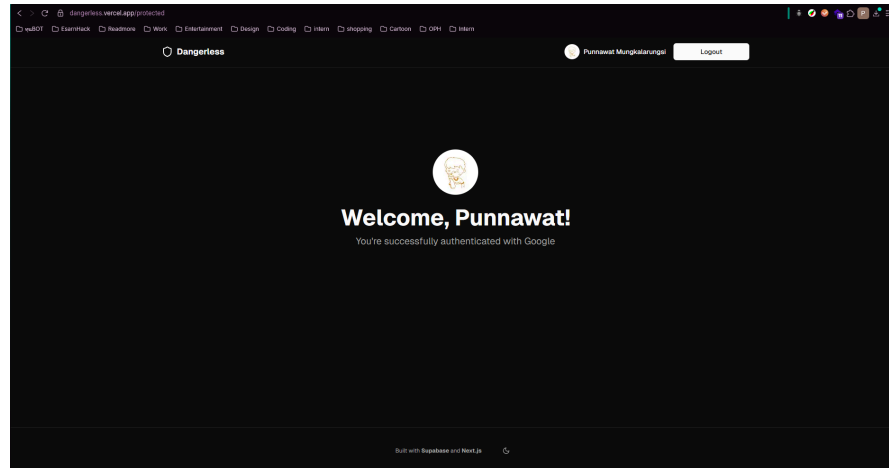
- Frontend: Vercel (edge-friendly, free tier OK for MVP)
- Backend: Supabase managed instance
- Testing/Quality: Playwright (e2e), Vitest + Testing Library (unit), ESLint/Prettier



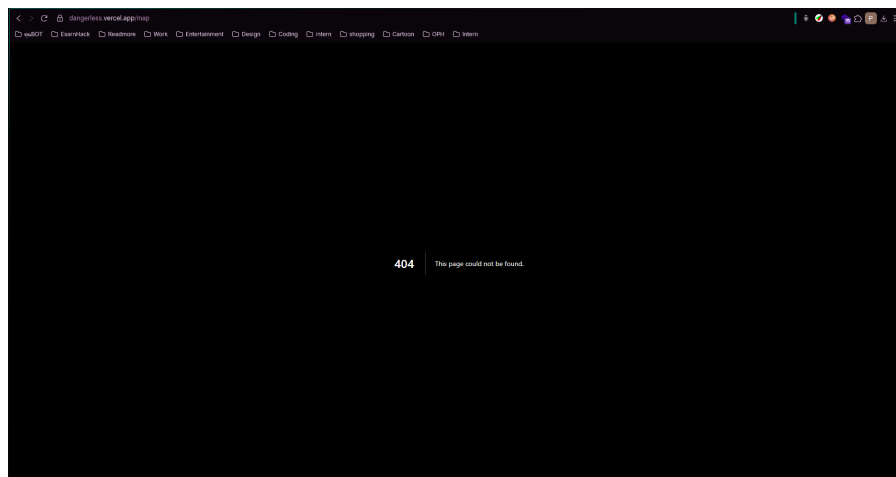
# Google OAuth Implementation

Repolink: <https://github.com/parunchxi/dangerless>

## Auth 200



## Auth 404



## Auth 401 (redirect to home page)

