

# Project Summary: Tofa

---

## Overview

Tofa is a modern e-commerce web application built with Next.js (App Router), TypeScript, Tailwind CSS, and Prisma ORM. It features robust authentication, a clean UI/UX, and a scalable backend with PostgreSQL. The project is ready for deployment on Vercel and uses Neon.tech for managed Postgres.

---

## Tech Stack

- **Frontend:** Next.js (App Router), TypeScript, Tailwind CSS
  - **Backend:** Next.js API routes, Prisma ORM
  - **Database:** PostgreSQL (Neon.tech)
  - **Authentication:** NextAuth.js (Google OAuth, credentials)
  - **UI Libraries:** Lucide icons, Radix UI, custom components
  - **Deployment:** Vercel
- 

## Folder Structure (Key Parts)

- `app/` — Main app pages, layouts, and API routes
  - `components/` — UI components (navbar, footer, product card, admin dashboard, etc.)
  - `contexts/` — React context providers (auth, cart)
  - `lib/` — Utility functions and Prisma client
  - `prisma/` — Prisma schema and migrations
  - `scripts/` — Database seed and init scripts
  - `public/` — Static assets
  - `styles/` — Global styles
- 

## Database Schema (Prisma)

- **User:** id, name, email, password, image, role (ADMIN/CUSTOMER), createdAt
  - **Product:** id, name, description, price, image, category, stock, createdAt
  - **Order:** id, userId, total, status, createdAt
  - **OrderItem:** id, orderId, productId, quantity, price
  - **CartItem:** id, userId, productId, quantity
  - **Settings:** id, key, value
  - **Enums:** Role, Category, OrderStatus
- 

## Authentication Flow

- **NextAuth.js** with Google OAuth and credentials (email/password)
- **Prisma Adapter** auto-creates users on first login
- **JWT sessions** for stateless auth

- **Custom login/register API** for credentials
- 

## Main Features

- User registration and login (Google, email/password)
  - Product listing, filtering, and details
  - Shopping cart (add, update, remove items)
  - Order placement and order history
  - Admin dashboard (manage products, orders, users)
  - Responsive, minimal UI/UX
- 

## Context Providers

- **AuthContext:** Handles user state, login, register, logout
  - **CartContext:** Manages cart state, add/remove/update items
- 

## UI/UX Libraries

- **Tailwind CSS:** Utility-first styling
  - **Radix UI:** Accessible UI primitives
  - **Lucide Icons:** Icon set
  - **Custom Components:** Navbar, footer, product card, admin sidebar, etc.
- 

## Deployment

- **Vercel:** For frontend and API hosting
  - **Neon.tech:** For managed PostgreSQL database
- 

## Environment Variables (.env)

- **DATABASE\_URL** — Postgres connection string
  - **NEXTAUTH\_URL** — App URL for NextAuth
  - **NEXTAUTH\_SECRET** — Secret for NextAuth JWT
  - **GOOGLE\_CLIENT\_ID** / **GOOGLE\_CLIENT\_SECRET** — Google OAuth
  - **EMAIL\_SERVER\_USER** / **EMAIL\_SERVER\_PASSWORD** — For email login (if used)
  - **STRIPE\_KEY** — For payment integration (if used)
- 

## Custom Logic & Notes

- **Seed scripts:** `scripts/seed.ts` seeds admin/customer users, products, orders, and settings
- **API routes:** Custom endpoints for auth, users, cart, products, orders
- **Error handling:** Prisma errors handled in API routes
- **UI/UX:** Modern, centered, floating header/footer, minimal product grid
- **Admin:** Separate admin pages for managing products, orders, users

---

## References

- See `ENVIRONMENT_SETUP.md` for environment setup
  - See `prisma/schema.prisma` for full database schema
  - See `scripts/seed.ts` for seed data
  - See `contexts/` for context providers
  - See `components/` for UI components
- 

*Generated on July 22, 2025*