WSB Social System Project Documentation

Version: 1.0.0
Date: 2026-01-20
Author: Antigravity (Assistant)

1. Introduction

This project is a modern Single Page Application (SPA) designed to serve as a professional social network (inspired by LinkedIn/ResearchGate). The application allows users to publish content, manage their network of connections, communicate in real-time, and interact with notifications.

2. Technology Stack

The application is built using modern web technologies:

- **Framework:** Next.js 14
- **Language:** TypeScript
- **UI Library:** React 18
- **Styling:** Tailwind CSS
- **Components:** Radix UI (primitives), Lucide React (icons)
- **State Management:** React Context API (AppContext)
- **Notifications:** Sonner (Toast system)

3. System Architecture

3.1 Directory Structure

- `/src/app` - Routing and Layouts (Next.js App Router).
- `/src/components` - User Interface (UI) components.
- `/src/contexts` - Business logic and global state.
- `/src/styles` - Global style sheets.

3.2 State Management (AppContext)

The core logic resides in `AppContext` (`src/contexts/AppContext.tsx`). It manages:

- **Users:** Current authorized user (`currentUser`) and the database of all users (`allUsers`).
- **Posts:** Feed array, handling likes, comments, and saving posts.
- **Notifications:** Activity notification system.
- **Messages:** Chat history and messaging logic.
- **Connections:** Logic for sending and accepting connection requests.

4. Key Functional Modules

4.1 Messaging Module (Messages)

- **File:** `src/components/messages/MessagesPage.tsx`
- **Description:** Full-screen chat interface.
- **Features:**

    - Conversation list with last message preview.
    - Contact search.
    - Dynamic chat window.
    - Real-time message sending handling (simulated via state updates).

4.2 Network & Connections

- **File:** `src/components/NetworkTab.tsx`
- **Description:** Managing relationships with other users.
- **Sections:**

    - **Connections:** List of active friends/connections.
    - **Invitations:** Received and sent connection requests.
    - **Suggestions:** People you may know (algorithm based on connection status).

4.3 Notifications

- **File:** `src/components/NotificationsTab.tsx`
- **Description:** Activity hub.
- **Event Types:** Likes, comments, shares, connection requests.
- **Features:** Mark as read (individually or bulk).

4.4 Feed & Posts (Home)

- **Components:** `PostCard.tsx`, `NewPostCard.tsx`
- **Description:** Main content feed. Supports rich media, PDFs, YouTube embeds.

5. Recent Fixes Report

5.1 Messaging Issue Fix

Problem: Newly created conversations did not allow sending the first message because the system could not identify the recipient before the `Conversation` object was physically created in the state array.
Solution: Modified the `getOtherParticipant` function in `MessagesPage.tsx`. Added fallback logic that, in the absence of an existing conversation, searches for the user in `allUsers` based on the predicted conversation ID.

6. Known Issues

1. **Connection System (Network):** The `ConnectionRequest` object does not store recipient information (`to`). This causes invitation filters to rely solely on the sender, which is a logic flaw to be addressed in future iterations.
2. **Notifications:** Lack of deep-linking (clicking a notification does not navigate to the content).