TrojanBook - Contact Management & Social Network

This project implements a contact management system called TrojanBook, evolving through three phases from a basic C++ application to a web-based system with recommendations.

Authors:

Owen Zeng owenzeng@usc.edu

James Wang jwang369@usc.edu

Xiaolei Yu xyu83571@usc.edu

Project Overview

TrojanBook allows users to manage contact information, establish friend connections, and (in Phase 3) interact through a web interface with friend recommendations.

Phases

- **Phase 1: Core C++ Implementation**
  - Focused on fundamental C++ classes (`Contact`, `Date`, `Person`, `Network`) for managing contacts using a doubly linked list.
  - Implemented file I/O (`networkDB.txt`) for persistence.
  - Provided a basic interactive text menu.
  - Details in README_P1.md
- **Phase 2: Friend Connections**
  - Enhanced the C++ application to support friend relationships between `Person` objects.
  - Introduced a unique `codeName` identifier for people.
  - Updated file I/O and the `Person` class to store/load friend connections.
  - Added menu options for connecting people and viewing sorted friend lists.
  - Details in README_P2.md
- **Phase 3: Web Interface & Recommendations**
  - Extended the C++ `Person` class to store additional key-value information (`std::map`).
  - Refactored the C++ application to be controlled via command-line arguments instead of an interactive menu.
  - Developed a Python script (`recommendations.py`) for content-based friend recommendations.
  - Built a Node.js/Express web application with a dynamic frontend (HTML/CSS/JS).

  - The Node.js backend acts as an interface, calling the compiled C++ executable for data operations and the Python script for recommendations.
  - Details in README_P3.md

How to Run (Phase 3 - Final Version)

1. **Compile C++ Code:** `bash make clean make test_network` This creates the `./test_network.o` executable used by the server.
2. **Install Node.js Dependencies (if needed):** `bash npm install express # Only required once`
3. **Run the Node.js Server:** `bash node server.js`
4. **Access Web UI:**
    - Open your browser and navigate to `http://localhost:3000` (or the port specified by the server output).

File Structure Overview

- **`.cpp` / `.h` files:** C++ source and header files for classes (`Person`, `Network`, `Contact`, `Date`, `misc`).
- **`test_*.cpp`:** C++ test files. `test_network.cpp` contains the `main` function for the command-line executable.
- **`networkDB.txt`:** The primary data file used by C++, Python, and Node.js.
- **`recommendations.py`:** Python script for generating friend recommendations.
- **`server.js`:** Node.js backend server (using Express).
- **`public/`:** Directory containing frontend files (`index.html`, `style.css`, `script.js`).
- **`package.json` / `package-lock.json`:** Node.js project files.
- **`Makefile`:** Used to compile the C++ code.
- **`README_P*.md`:** Detailed documentation for each project phase.