Connect MVP SRS

Okay, here's a draft of a Software Requirements Specification (SRS) document for the "Connect" app, based on our previous conversations and the mock-up you provided. This is a starting point, and you'll need to fill in the details specific to your project:

**Software Requirements Specification: Connect App**

**1. Introduction**

* **1.1 Purpose:**
  + This document specifies the requirements for the "Connect" application, a mobile/web app designed to help users manage and track interactions with their contacts, facilitating stronger relationships and better record-keeping.
* **1.2 Scope:**
  + This document covers the functional, non-functional, and interface requirements for the Connect app.
* **1.3 Intended Audience:**
  + This document is intended for the development team, testers, project managers, and stakeholders involved in the Connect app project.

**2. Overall Description**

* **2.1 Product Perspective:**
  + Connect is a standalone application that provides a centralized hub for contact management, communication, and activity logging. It aims to provide a more organized and personal alternative to standard contact lists and scattered communication methods.
* **2.2 Product Functions:**
  + Contact Profile Management (create, read, update, delete)
  + Messaging (separate and all messages views)
  + Journal/Activity Log
  + Keyword Search
* **2.3 User Characteristics:**
  + Professionals (sales, recruiting, consulting) who rely on relationship building.
  + Project managers tracking team interactions.
  + Individuals seeking better relationship management.
  + Tech-literate users comfortable with mobile apps.
* **2.4 Operating Environment:**
  + **Mobile:**
    - iOS (latest 3 versions)
    - Android (latest 3 versions)
  + **Web:**
    - Latest versions of Chrome, Firefox, Safari, Edge
  + **Connectivity:** Requires internet connectivity for most functionalities. Offline access for viewing previously loaded contact information may be considered.
* **2.5 Design and Implementation Constraints:**
  + **Data Privacy:** Compliance with GDPR, CCPA, and other relevant privacy regulations.
  + **Security:** Secure storage of user data and secure communication.
  + **Scalability:** The system should be designed to handle a growing number of users and contacts.
  + **Technology Stack:** [*To be determined based on your choices - e.g., React Native, Node.js, MongoDB, Firebase*]

**3. Specific Requirements**

* **3.1 Functional Requirements**
  + **3.1.1 Contact Management**
    - **FR.1.1 Create Contact:** Users shall be able to create new contact profiles with the following fields:
      * Full Name (Text, Required)
      * Profile Picture (Image Upload, Optional)
      * Phone Number (Text, Optional, Validated format)
      * Facebook Link (URL, Optional, Validated URL format)
      * Email Address (Text, Optional, Validated email format)
      * LinkedIn Link (URL, Optional, Validated URL format)
      * "View Information" (Expandable section with customizable fields):
        + Job Title (Text, Optional)
        + Company (Text, Optional)
        + Notes (Text Area, Optional)
        + [*Add other customizable fields here, specifying data type for each*]
    - **FR.1.2 Read Contact:** Users shall be able to view existing contact profiles with all associated information.
    - **FR.1.3 Update Contact:** Users shall be able to modify existing contact profiles.
    - **FR.1.4 Delete Contact:** Users shall be able to delete contact profiles. A confirmation prompt shall be displayed before deletion.
    - **FR.1.5 Contact Grouping/Tagging:**
      * Users shall be able to assign tags to contacts (e.g., "Client", "Prospect", "Friend").
      * Users shall be able to filter contacts by tag.
    - **FR.1.6 Import Contacts:**
      * Users shall be able to import contacts from a CSV file.
      * The system shall support mapping CSV columns to contact fields.
    - **FR.1.7 Export Contacts:**
      * Users shall be able to export contacts to a CSV file.
      * The system shall allow users to select which fields to export.
  + **3.1.2 Messaging**
    - **FR.2.1 Send Message:** Users shall be able to send text-based messages to contacts.
    - **FR.2.2 Receive Message:** Users shall be able to receive text-based messages from contacts.
    - **FR.2.3 Separate Message View:** Users shall be able to view the message history for a specific contact.
    - **FR.2.4 All Messages View:** Users shall be able to view all messages in a combined inbox, sorted by date/time.
    - **[Future] FR.2.5 Message Attachments:** Users shall be able to send and receive attachments (images, documents).
    - **[Future] FR.2.6 Read Receipts:** The system shall display read receipts for sent messages.
  + **3.1.3 Journal/Activity Log**
    - **FR.3.1 Add Entry:** Users shall be able to add new journal entries to a contact's profile.
      * Each entry shall include:
        + Date/Time (Automatically generated)
        + Text-based notes (Required)
    - **FR.3.2 Display Entries:** Journal entries shall be displayed in chronological order (newest first).
    - **FR.3.3 Search Entries:** Users shall be able to search journal entries for a contact using keywords.
    - **[Future] FR.3.4 Categorize/Tag Entries:** Users shall be able to assign categories/tags to journal entries.
  + **3.1.4 Search**
    - **FR.4.1 Global Search:** Users shall be able to search across all contact profiles, journal entries, and messages using keywords.
    - **FR.4.2 Search Logic:**
      * The search shall be case-insensitive.
      * The search shall support both AND and OR logic (specify which one to use by default, or allow the user to choose).
      * [*Decide if you will implement partial word matching*]
    - **FR.4.3 Search Results:**
      * Search results shall display the contact name and a snippet of the matching content, along with the source (e.g., "Job Title", "Journal Entry").
      * Matching keywords shall be highlighted in the search results.
  + **3.1.5 User Authentication**
    - **FR.5.1 Registration:** New users shall be able to create an account by providing an email address and password.
    - **FR.5.2 Login:** Existing users shall be able to log in using their email address and password.
    - **FR.5.3 Social Login:** Users shall be able to log in using their existing social media accounts (Facebook, Google, LinkedIn).
    - **FR.5.4 Password Reset:** Users shall be able to reset their password if they forget it.
* **3.2 Non-Functional Requirements**
  + **3.2.1 Performance**
    - The app shall load contact profiles within 2 seconds.
    - Search results shall be displayed within 3 seconds.
    - The app shall be able to handle at least 10,000 users concurrently.
  + **3.2.2 Security**
    - User passwords shall be stored securely using a one-way hashing algorithm.
    - All communication between the app and the server shall be encrypted using HTTPS.
    - The app shall be protected against common web vulnerabilities (e.g., XSS, SQL injection).
  + **3.2.3 Usability**
    - The app shall have an intuitive and user-friendly interface.
    - The app shall be accessible to users with disabilities (WCAG compliance).
  + **3.2.4 Reliability**
    - The app shall have an uptime of 99.9%.
    - The app shall handle errors gracefully and provide informative error messages.
  + **3.2.5 Maintainability**
    - The codebase shall be well-documented and modular.
    - The app shall be designed to be easily updated and maintained.
* **3.3 External Interface Requirements**
  + **3.3.1 Social Media APIs:**
    - The app shall integrate with the Facebook, Google, and LinkedIn APIs for social login and contact information (with user permission).
    - [*Specify the specific API endpoints to be used*]
  + **3.3.2 Backend API:**
    - The app shall communicate with a backend API to retrieve and store data.
    - The API shall use JSON for data exchange.
    - [*Provide a separate API documentation document*]

**4. System Models**

* [*Include use case diagrams, data flow diagrams, and entity-relationship diagrams (ERD) to visually represent the system*]

**5. Future Considerations**

* Push notifications for new messages and reminders.
* Calendar integration.
* Relationship strength indicator.
* Data analytics dashboards.

**6. Appendix**

<https://kzmq7zuedcoptw0ezl1v.lite.vusercontent.net/>

**Remember to:**

* **Fill in the bracketed sections** with specific details relevant to your project.
* **Review and revise** this document with your stakeholders to ensure that it accurately reflects the requirements for the Connect app.
* **Keep the SRS updated** throughout the development process as requirements change.
* **Consider using a template:** There are many SRS templates available online that can help you structure your document.

This SRS will serve as a solid starting point for your development team. Good luck!

**Citations:**

1. <https://pplx-res.cloudinary.com/image/upload/v1740335760/user_uploads/QwTvwpSNQpqOyrB/Mock3.jpg>
2. <https://pplx-res.cloudinary.com/image/upload/v1740335945/user_uploads/aEbbdDCdEURhlPX/image.jpg>
3. <https://pplx-res.cloudinary.com/image/upload/v1740335947/user_uploads/SZZjpMCIZuyXzDZ/image.jpg>
4. <https://pplx-res.cloudinary.com/image/upload/v1740336022/user_uploads/mcbbJXDNbaajpXV/image.jpg>