CREATE a web application using good practices to split the resources and organize files and folders.   
  
Application’s name is Staketrack.   
  
Use at minimal: Component-Based Architecture, MVC Pattern, Event-Driven Communication, Separation of Concerns, Proper model classes with encapsulation, Repository pattern for data operations, Dedicated services for storage and import/export.  
  
**For implementation use:  
- Native Web Technologies**: Uses vanilla JavaScript, HTML, and CSS with no external dependencies.  
- **Modern JavaScript Features**: Leverages ES6+ features like classes, modules, and arrow functions.   
- **Responsive Design**: CSS variables and flexible layout adapt to different screen sizes. -- **Accessibility Considerations**: Proper labels, semantic HTML, and keyboard navigation support.   
- **Performance Optimization**: Event delegation, efficient DOM manipulation, and minimal repaints.

**- Responsive Design**: Works on both desktop and mobile devices.

**- Backend Services:** FIREBASE services and capabilities for deployment and backend.   
  
The application should support the following functional capabilities:

1. **Comprehensive Stakeholder Profiles**: Each stakeholder can be documented with all the attributes from our framework:
   * Name/Role
   * Influence and Impact ratings (1-10)
   * Relationship quality rating (1-10)
   * Primary interests
   * Potential contribution
   * Risk if disengaged
   * Communication style preferences
   * Engagement strategy
   * Measurement approach
   * Stakeholder category
   * Interaction Log
2. For each field provide tooltips with a description and explanation with an example to help users.
3. **Visual Influence-Impact Matrix:** The application provides an interactive quadrant map where stakeholders are plotted based on their influence and impact levels.

quadrantChart

title Stakeholder Influence-Impact Matrix

x-axis Low Influence --> High Influence

y-axis Low Impact --> High Impact

quadrant-1 "Key Players" "High Impact / High Influence"

quadrant-2 "Meet Their Needs" "High Impact / Low Influence"

quadrant-3 "Show Consideration" "Low Impact / Low Influence"

quadrant-4 "Keep Satisfied" "Low Impact / High Influence"

1. **Visual Indicators**: Stakeholders are color-coded based on relationship quality (red for weak, orange for medium, green for strong) to provide quick visual cues about relationship health.
2. Plot stakeholders with a mouseover capability to display name/role field.
3. “Interaction Log” functionality is a journal system to log on an open text format multiple interaction records:
   * Keep track of data and time of the journal entry as a metadata
   * Visualization as a single pop-up window when requested.
4. For each stakeholder provide a way to invoke an action to request ADVICE & IDEAS
   * This action will trigger a call to an LLM service
   * Build the prompt as a template system with INSTRUCTION, CONTEXT, and OUTPUT FORMAT
   * For CONTEXT build a LLM-friendly view of the selected stakeholder record, including the interaction journal records.
   * It will be using the Anthropic’s API. You can use a LLM abstraction library to implement it.
   * Output should be displayed as a pop-up respecting formatting with a standard annotation approach (provided as OUTPUT FORMAT)
5. For the MAP, provide an action button to request ADVISE NEXT BEST ACTION
   * This action will trigger a call to an LLM service
   * Build the prompt as a template system with INSTRUCTION, CONTEXT, and OUTPUT FORMAT
   * For CONTEXT build a LLM-friendly view of all stakeholder records, including the last 3 interaction journal records of each stakeholder.
   * It will be using the Anthropic’s API. You can use a LLM abstraction library to implement it.
   * Output should be displayed as a pop-up respecting formatting with a standard annotation approach (provided as OUTPUT FORMAT)
6. **Data Persistence**: The tool automatically saves data to user’s browser's local storage, ensuring they don't lose the work between sessions. And persists it on the backend if the user is logged in.
7. **Import/Export Functionality**: You can export your stakeholder data to share with others or back it up, and import previously saved data.
8. **Stakeholder List View**: Provides an alternative way to view and select stakeholders with their key attributes at a glance.
9. **Provide tooltips to guide users**
10. User Authentication
    * includes a flexible authentication system with: Multiple sign-in options (Google, Microsoft, and email/password), Persistent user profiles, Secure access control to user data
    * Session management with sign-out capability
    * User login is optional. Anonymous users are still supported.
11. Multiple Maps Management
    * Create and name multiple stakeholder maps
    * Switch between different maps via a dropdown selector
    * Edit map metadata (name, description)
    * Delete maps they no longer need
    * See metadata such as update timestamps and stakeholder counts
12. Cloud Data Persistence
    * All data is stored in Firebase Firestore, providing: Real-time data synchronization, Secure data storage with proper access controls, Structured document-based storage with collections for users, maps, and stakeholders, Database rules that enforce data access security, Optimized queries with proper indexing
13. Analytics Integration
    * Collect anonymous usage data to help administrators understand:
    * User engagement patterns
    * Feature popularity
    * Stakeholder distribution trends
    * Session duration and frequency
    * Common errors and pain points