

Project Title: Library Management

Problem Statement

Managing and tracking books can become cumbersome when done manually or on paper. Small libraries, students, or hobbyists often need a lightweight system to record book details, view them quickly, and make edits without installing heavy software or maintaining a server. The Simple Library Management website addresses this by offering a browser-based CRUD tool using only HTML, CSS, and JavaScript with localStorage.

Requirement Gathering

- Identify what core actions the user needs (add, view, edit, delete books).
- Decide which fields are mandatory (Title, Author, Year, ISBN).
- Determine storage mechanism (localStorage vs. server).
- Collect UI expectations (form + table layout, buttons).
- Plan file structure (index.html, styles.css, script.js).
- Document non-functional needs (works offline, simple design).

Stakeholder Analysis

- Define primary users (students, small libraries, demo users).
- Identify secondary users (developers learning CRUD apps).
- Note stakeholders' technical skills (beginner-friendly, no server).
- Capture feedback on desired features (extra fields, pop-ups).
- Understand constraints (browser-only, small data sets).
- Set expectations for future enhancements (server API, multi-user).

Business Process Mapping

- Map the flow of adding a book (form input → JS → localStorage → table).
- Describe how data persists after refresh (JSON in localStorage).
- Show the edit flow (click edit → prompt/update → save back).
- Map the delete flow (click delete → confirm → remove from storage).
- Define initial data state (empty table).
- Document error handling or confirmations (validation, clearing storage).

Industry-Specific Use Case Analysis

- Compare with small-scale library or personal book lists.
- Explore educational uses (teaching CRUD in classes).
- Consider adaptation for inventory or asset tracking.
- Assess scalability limits (localStorage size, single user).
- Identify privacy and data safety considerations (local only).
- Evaluate user interface needs across devices (desktop, mobile).

AppExchange / Expansion Exploration

- Plan how to extend fields (Genre, Publisher, Ratings).
- Replace prompts with inline edit forms or modals.
- Investigate moving from localStorage to a simple backend or API.
- Consider packaging as a small open-source app or template.
- Experiment with new UI styles (Tailwind, Bootstrap, animations).
- Explore publishing on platforms like Salesforce AppExchange or GitHub Pages as a free tool.