Software Design Document (SDD)

Version: Want2Remember (v4)

Project Name: Want2Remember

Snapshot: 4 (Final Checkpoint)

Team Members: Kevin Bayona-Galindo & Nikolazi Tartinsky

**1. Introduction Purpose:**

This version finalizes the system design and includes the final UI styling improvements. It reflects the completed implementation of memory CRUD operations (Create, Read, Delete). Audience: Instructor, developers, and evaluators

**2. System Architecture Frontend:**

HTML/CSS/JavaScript (Vanilla) Backend: None (data stored client-side) Data Storage: Browser-based localStorage

**3. Component Overview - Memory Input Form:**

User enters a memory (title + note) - Memory List Display: Shows all stored memories dynamically - Delete Button: Allows individual memory removal - Style Layer: Dark gray background with clean, centered layout and white content cards

**4. Workflow Summary:**

1. User opens `index.html`

2. Existing memories load from localStorage

3. User adds a memory

4. Memory is saved to localStorage and displayed in the list

5. User can delete any memory

6. UI and storage update immediately

**5. UI Details:**

- Form includes labeled fields and a styled submit button

- Memory cards use box shadow, padding, and spacing for readability

- The list is styled using Flexbox with spacing between cards

**6. Glossary - DOM:**

- DOM: Document Object Model

- localStorage: Browser API for persistent key-value data

- CRUD: Create, Read, Update (future), Delete

**7. References MDN Web Docs:**

<https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>

<https://docs.docker.com/get-started/>

<https://hub.docker.com/_/nginx>

<https://www.overleaf.com/learn>

<https://www.gurock.com/testrail/docs>

<https://www.atlassian.com/software/jira>