

# Enchiridion Roadmap

This document outlines potential improvements, new features, code cleanup opportunities, and technical debt for the Enchiridion project - a terminal novel writing assistant with AI integration for Lean 4.

---

## Feature Proposals

### [Priority: High] Auto-Save Functionality

**Description:** Implement the auto-save feature that is already configured but not implemented.

**Rationale:** The Config structure already has `autoSaveEnabled` and `autoSaveIntervalMs` fields, but the application does not use them. Auto-save is critical for a writing application to prevent data loss.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/App.lean`  
- `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean`

**Estimated Effort:** Medium

**Dependencies:** None

---

### [Priority: High] Project Load Dialog

**Description:** Add the ability to load existing projects from disk, not just save.

**Rationale:** Currently the application only supports saving projects and always starts with a sample project. Users need to be able to open their previously saved work.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/App.lean` -  
`/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean` - `/Users/Shared/Projects/lean-`  
(AppMode already has `.loading`)

**Estimated Effort:** Medium

**Dependencies:** None

---

### [Priority: High] New Project Creation Dialog

**Description:** Add ability to create a new project with custom title and author from the UI.

**Rationale:** Currently the application always starts with a hardcoded sample project (“The Great Adventure”). Users should be able to create new projects with their own titles.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/App.lean` -  
`/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean` - `/Users/Shared/Projects/lean-`

**Estimated Effort:** Medium

**Dependencies:** None

---

### [Priority: Medium] Character Relationship Tracking

**Description:** Add the ability to define and track relationships between characters.

**Rationale:** Novel writing often requires tracking complex character relationships. The current Character model has basic traits but no relationship data.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Character.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Json.lean - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Character.lean (Notes panel)

**Estimated Effort:** Medium

**Dependencies:** None

---

#### [Priority: Medium] Scene Outline/Summary View

**Description:** Add a bird's-eye view showing all scene synopses in a condensed format.

**Rationale:** Writers often need to see the overall story structure at a glance. The current navigation only shows titles, not synopses.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/Focus.lean (new panel/mode) - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Layout.lean

**Estimated Effort:** Medium

**Dependencies:** None

---

#### [Priority: Medium] Chapter/Scene Reordering

**Description:** Allow drag-and-drop style reordering of chapters and scenes via keyboard.

**Rationale:** Story structure often changes during writing. Currently there is no way to reorder chapters or scenes without deleting and recreating them.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean

**Estimated Effort:** Medium

**Dependencies:** None

---

#### [Priority: Medium] Scene/Chapter Rename Inline

**Description:** Allow renaming chapters and scenes directly in the navigation panel.

**Rationale:** Currently chapters and scenes are created with generic names ("Chapter 1", "Scene 1") and there is no way to rename them. Users need to be able to give meaningful titles.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean

**Estimated Effort:** Small

**Dependencies:** None

---

#### [Priority: Medium] Undo/Redo Stack

**Description:** Implement undo/redo functionality for text editing and structural changes.

**Rationale:** Essential for any text editor. Currently there is no way to undo accidental deletions or changes.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean - Potentially requires changes to terminus TextArea

**Estimated Effort:** Large

**Dependencies:** May require terminus enhancements

---

#### [Priority: Medium] AI Provider Selection

**Description:** Support multiple AI providers beyond OpenRouter (e.g., direct OpenAI, Anthropic, local models via Ollama).

**Rationale:** Users may prefer different AI providers for cost, privacy, or capability reasons.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/AI/OpenRouter.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Config.lean - New files for each provider

**Estimated Effort:** Large

**Dependencies:** None

---

#### [Priority: Medium] Chat History Persistence

**Description:** Save and restore AI chat history with the project.

**Rationale:** Chat history provides context for ongoing AI conversations but is lost when the application restarts.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Project.lean -  
/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Json.lean

**Estimated Effort:** Medium

**Dependencies:** None

---

#### [Priority: Medium] Search and Replace

**Description:** Implement find and replace functionality across the current scene or entire novel.

**Rationale:** Essential editing feature for any writing application.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean - /Users/Shared/Projects/lean

**Estimated Effort:** Medium

**Dependencies:** None

---

#### [Priority: Low] Word Goal Tracking

**Description:** Allow users to set daily/session word count goals with progress tracking.

**Rationale:** Many writers use word count goals to maintain productivity. The infrastructure exists (word count calculation) but no goal tracking.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Config.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean -  
/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean

**Estimated Effort:** Small

**Dependencies:** None

---

#### [Priority: Low] Multiple Export Formats

**Description:** Support export to additional formats beyond Markdown (e.g., DOCX, EPUB, plain text).

**Rationale:** Different publishing workflows require different formats.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/State/AppState.lean  
(exportToMarkdown) - New export modules

**Estimated Effort:** Large

**Dependencies:** Would require external libraries or FFI

---

#### [Priority: Low] Theme Support

**Description:** Allow customization of UI colors and styles.

**Rationale:** Writers often prefer different color schemes, especially for long writing sessions.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Config.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean

**Estimated Effort:** Medium

**Dependencies:** None

---

#### [Priority: Low] Plugin System

**Description:** Allow extensions to add custom AI prompts, export formats, or UI panels.

**Rationale:** Different writers have different workflows; a plugin system would allow customization.

**Affected Files:** Architecture-level change

**Estimated Effort:** Large

**Dependencies:** None

---

### Code Improvements

#### [Priority: High] Proper Timestamp Implementation

**Current State:** `Timestamp.now` uses `IO.monoNanosNow` which returns monotonic time, not Unix epoch time. The conversion to milliseconds is also incorrect.

**Proposed Change:** Use proper system time for timestamps. The comment says “milliseconds since Unix epoch” but the implementation uses monotonic nanoseconds.

**Benefits:** Correct timestamps for file metadata and display.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Types.lean  
(lines 39-43)

**Estimated Effort:** Small

---

#### [Priority: High] Error Handling Improvements

**Current State:** Many operations silently fail or return empty results. For example, `Config.loadFromFile` catches all exceptions and returns `none`.

**Proposed Change:** Use proper error types with descriptive messages. Consider a unified Result monad or Except-based error handling.

**Benefits:** Better debugging, clearer error messages for users.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Config.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Storage/FileIO.lean -  
/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/AI/OpenRouter.lean

**Estimated Effort:** Medium

---

#### [Priority: Medium] Refactor Update Function

**Current State:** The `update` function in `UI/Update.lean` is a large chain of if-else statements that is difficult to maintain.

**Proposed Change:** Consider using pattern matching more extensively, or a command pattern to decouple key handling from actions.

**Benefits:** Better maintainability, easier to add new shortcuts.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean

**Estimated Effort:** Medium

---

#### [Priority: Medium] Consolidate JSON Serialization

**Current State:** Each model type has manual `ToJson/FromJson` instances with repetitive boilerplate.

**Proposed Change:** Consider using deriving for JSON instances where possible, or create helper macros to reduce boilerplate.

**Benefits:** Less code duplication, fewer serialization bugs.

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Json.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Novel.lean - /Users/Shared/Projects/  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/WorldNote.lean -  
/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Project.lean

**Estimated Effort:** Medium

---

#### **[Priority: Medium] Separate Streaming Logic**

**Current State:** AI streaming is handled inline in `runLoop` which makes the function very long and complex.

**Proposed Change:** Extract streaming management into a dedicated module or state machine.

**Benefits:** Cleaner code, easier to test streaming behavior.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/App.lean`  
- `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/AI/Streaming.lean`

**Estimated Effort:** Medium

---

#### **[Priority: Medium] Type-Safe Panel Focus Handling**

**Current State:** Panel-specific update functions are called based on enum matching. The current approach works but could be more type-safe.

**Proposed Change:** Consider using typeclasses or a more structured approach to panel handling.

**Benefits:** Compile-time guarantees that all panels are handled correctly.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Update.lean`  
- `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean`

**Estimated Effort:** Medium

---

#### **[Priority: Low] Word Count Algorithm Improvement**

**Current State:** Word count uses simple space splitting which may be inaccurate for edge cases.

**Proposed Change:** Implement a more robust word counting algorithm that handles punctuation, multiple spaces, etc.

**Benefits:** More accurate word counts.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Novel.lean`  
(`Scene.updateWordCount`)

**Estimated Effort:** Small

---

#### **[Priority: Low] Configurable Layout**

**Current State:** Panel sizes are hardcoded percentages in `LayoutConfig`.

**Proposed Change:** Allow users to resize panels via keyboard or configuration.

**Benefits:** Better user experience for different screen sizes and preferences.

**Affected Files:** - `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Layout.lean`  
- `/Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Config.lean`

**Estimated Effort:** Medium

---

## Code Cleanup

### [Priority: High] Remove Legacy PromptType

**Issue:** PromptType enum in AI/Prompts.lean is marked as legacy with a comment “use AIWritingAction instead” but is still present and used by buildPrompt.

**Location:** /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/AI/Prompts.lean (lines 83-115, 118-139)

**Action Required:** 1. Remove the PromptType enum 2. Remove the buildPrompt function that uses it 3. Ensure all callers use buildWritingActionPrompt instead

**Estimated Effort:** Small

---

### [Priority: Medium] Duplicate parseStringArray Functions

**Issue:** The helper function parseStringArray is defined identically in both Character.lean and WorldNote.lean.

**Location:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Character.lean (lines 73-77) - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/WorldNote.lean (lines 114-118)

**Action Required:** Move the function to Core/Json.lean and export it for shared use.

**Estimated Effort:** Small

---

### [Priority: Medium] Unused Config Fields

**Issue:** autoSaveEnabled and autoSaveIntervalMs config fields are parsed and saved but never used.

**Location:** /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Core/Config.lean (lines 17-18)

**Action Required:** Either implement auto-save feature or remove the unused fields.

**Estimated Effort:** Small (removal) or Medium (implementation)

---

### [Priority: Medium] Missing File Existence Check

**Issue:** Storage.fileExists reads the entire file to check if it exists, which is inefficient.

**Location:** /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Storage/FileIO.lean (lines 67-72)

**Action Required:** Use IO.FS.metadata or similar to check file existence without reading content.

**Estimated Effort:** Small

---

### [Priority: Low] Inconsistent Error Message Formatting

**Issue:** Error messages throughout the codebase have inconsistent formatting (some with prefixes, some without).

**Location:** Multiple files in AI and Storage modules

**Action Required:** Standardize error message format, possibly with a central error type.

**Estimated Effort:** Small

---

#### [Priority: Low] Magic Numbers in Draw.lean

**Issue:** Several hardcoded values in the draw functions (e.g., popup sizes, column widths).

**Location:** /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/UI/Draw.lean

**Action Required:** Extract magic numbers into named constants.

**Estimated Effort:** Small

---

#### [Priority: Low] Test Coverage for UI Logic

**Issue:** While model and state tests are comprehensive, UI update logic has no direct tests.

**Location:** /Users/Shared/Projects/lean-workspace/enchiridion/Tests/Main.lean

**Action Required:** Add tests for `updateNavigation`, `updateEditor`, `updateChat`, `updateNotes` functions.

**Estimated Effort:** Medium

---

#### [Priority: Low] Missing Documentation Comments

**Issue:** Some public functions lack documentation comments explaining their purpose and usage.

**Location:** Various files, particularly in UI modules

**Action Required:** Add doc comments to all public functions.

**Estimated Effort:** Small

---

## API Enhancements

#### [Priority: Medium] Builder Pattern for Project Creation

**Description:** Currently creating a project with all optional fields requires many struct updates. A builder pattern would improve ergonomics.

**Example:**

```
-- Current
let novel := { novel with genre := "Fantasy", synopsis := "..." }
```

```
-- Proposed
let novel := Novel.builder "Title" |>.author "Name" |>.genre "Fantasy" |>.build
```

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Novel.lean  
- /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Project.lean

**Estimated Effort:** Small

---



### [Priority: Low] Query DSL for Project Content

**Description:** Add a simple query DSL for finding content in the project (e.g., scenes containing a character name, notes by category).

**Example:**

```
project.findScenes (fun s => s.content.containsSubstr "Sarah")
project.findNotes (.category == .location)
```

**Affected Files:** - /Users/Shared/Projects/lean-workspace/enchiridion/Enchiridion/Model/Project.lean

**Estimated Effort:** Small

---

### Technical Debt Summary

| Category     | High     | Medium    | Low       |
|--------------|----------|-----------|-----------|
| Features     | 3        | 8         | 4         |
| Improvements | 2        | 5         | 2         |
| Cleanup      | 2        | 3         | 4         |
| API          | 0        | 1         | 1         |
| <b>Total</b> | <b>7</b> | <b>17</b> | <b>11</b> |

---

### Recommended Next Steps

- 1. Immediate (High Priority):**
  - Remove legacy `PromptType` enum (cleanup)
  - Fix timestamp implementation (improvement)
  - Implement project load functionality (feature)
- 2. Short-term (Medium Priority):**
  - Implement auto-save using existing config (feature)
  - Add scene/chapter renaming (feature)
  - Consolidate duplicate code (cleanup)
- 3. Long-term (Low Priority):**
  - Consider undo/redo implementation
  - Explore plugin system architecture
  - Add comprehensive UI testing