Feature & Requirements Specification: Content Creator's Assistant Application

Introduction

This document details the features and functional requirements for a sophisticated "Content Creator's Assistant" application. The specifications outlined herein are derived from a successful and feature-rich prototype built using a long-running agent harness. This document serves as a comprehensive blueprint for a production-quality implementation to be executed by an advanced coding agent, ensuring all critical functionality and user experience elements are captured.

1.0 Core Application & Project Management

1.1. Introduction to Core Functionality

This section outlines the application's foundational features, which are architected to provide a stable and robust harness for the agentic content creation workflows. Unlike unstable, single-shot implementations, these requirements for the user interface, project lifecycle management, and system configuration form a resilient backbone for the user experience. A polished, intuitive, and reliable core application is critical for user adoption and for enabling the efficient management of long-running, stateful tasks.

1.2. General UI/UX and Theming

The application shall provide a professional and user-centric interface that prioritizes clarity and ease of use.

- **Dual-Mode Theming:** The application shall support both a light mode and a dark mode. The user shall be able to toggle between these two themes at any time to suit their preference and working environment.
- **Polished Interface:** The overall user interface design shall be professional, polished, and intuitive. This standard is in direct contrast to simplistic, proof-of-concept implementations and is a core requirement for the final product.
- **Workflow Progress Indicator:** A persistent visual element shall be present within the content generation view to clearly indicate the user's current stage in the workflow (e.g., Hooks, Intros, Titles, Thumbnails).

1.3. Project Lifecycle Management

The main dashboard shall provide comprehensive project management capabilities. These features provide the necessary scaffolding for managing multiple, long-running content generation tasks. The ability to filter, duplicate, and edit projects is crucial for allowing both the user and the system to manage complex workflows efficiently.

- **Project Dashboard:**
- **Global Actions:** A global "Create New Project" button shall be present on the main dashboard. The creation process shall prompt the user for the following required details:
- Project Name
- Video Topic
- Target Audience

- Content Style
- **Card-Based Display:** All projects shall be represented as individual cards on the main dashboard screen, providing an at-a-glance overview.
- **Status Filtering:** The interface shall include a mechanism for users to filter the displayed projects based on their status.
- **Project Card Actions:** When a user hovers over a project card, a set of action icons (e.g., Open, Edit, Duplicate, Delete) shall become available. The following actions are required:
- **Open Project:** This primary action allows the user to enter the main content creation workflow for the selected project.
- **Edit Project:** The system shall enable the user to modify the project's core details (Name, Topic, etc.) after its initial creation.
- **Duplicate Project:** The application shall provide a feature to create an exact copy of an existing project, preserving all its details and generated content.
- **Delete Project:** A fully operational function to permanently remove a project from the system shall be implemented.

1.4. System Configuration

The application shall grant users advanced control over the AI's output by allowing for the direct modification of system prompts.

- **Settings Interface:** A dedicated "Settings" area shall be accessible from the main dashboard.
- **Editable System Prompts:** Within the settings interface, the application shall provide distinct text areas where users can view and edit the system prompts for each of the AI agents responsible for content generation:
- Hooks Agent
- Intros Agent
- Titles Agent
- Thumbnails AgentSatisfying these core requirements is foundational to delivering a stable and intuitive user experience, upon which the primary content generation workflow is built.

2.0 Content Generation Workflow

2.1. Introduction to the Generation Workflow

The multi-step content generation workflow is the application's core value-delivery mechanism. It is architected to overcome the context loss and premature completion failures observed in single-shot agent implementations. By enforcing a guided, step-by-step process with persistent state, we ensure a fluid and non-destructive creative process, mirroring the incremental and test-driven approach of an effective software engineering lifecycle.

2.2. Interactive Content Generation UI

The interface for generating and managing text-based ideas (hooks, intros, titles) shall be dynamic and user-friendly.

- **Card-Based Idea Presentation:** All generated ideas for hooks, intros, and titles shall be displayed as individual, self-contained cards.

- **Direct Card Interactions:** Each card shall be interactive and support the following direct user actions:
- **Copy Text:** A button or icon to instantly copy the card's text content to the user's clipboard.
- **Edit Text:** An inline editing feature that allows the user to click and directly modify the generated text on the card itself.
- **Select Item:** A clear mechanism (e.g., a "Select" button on the card, or making the entire card clickable) to mark it as the chosen asset for that workflow stage.
- **Workflow Controls:** The generation interface shall provide global controls for managing the set of ideas:
- **Regenerate All:** A function to discard the current set of generated ideas and prompt the AI to generate a new batch from scratch.
- **Add More:** A function to generate additional ideas and append them to the existing list, without discarding the current set.

2.3. Thumbnail Generation Module

The thumbnail generation module is a highly interactive and complex feature that goes beyond simple text generation to create final visual assets.

- **Core Functionality:**
- **Image Generation Engine:** The system shall utilize the Nano Banana model to generate actual image thumbnails, not text-based descriptions.
- **Input & Asset Management:**
- **Reference Image Upload:** The system shall enable the user to upload a primary reference image, such as a photo of the content creator.
- **Logo/Asset Upload:** The system shall enable the user to upload supplementary assets, such as brand logos or icons, to be incorporated into the thumbnail designs.
- **Output & Refinement Interface:**
- **Generated Thumbnail Display:** The interface shall display the generated thumbnail images in a grid layout for easy comparison and selection.
- **Interaction Options:** For each generated thumbnail, the following interactive tools shall be available:
- **View Full Screen:** An option to view an enlarged, lightbox version of the selected image.
- **View Prompt:** A feature that allows the user to see the exact text prompt that was sent to the Nano Banana model to create that specific image.
- **Upscale Image:** A function to process the selected thumbnail and increase its resolution to 4K.
- **Download Image:** An option to download the final, upscaled image file to the user's local machine.
- **Refine Image:** An interactive, incremental refinement loop where the user can provide a natural language text prompt to modify the existing image (e.g., "change the text to 'Beginner Tutorial'").
- **Versioning and History:**
- **Revision Tracking:** When an image is modified using the "Refine" feature, the new version shall be displayed, and the original version shall be preserved in a history log.
- **History Viewer:** A "History" button shall appear on any refined thumbnail, allowing the user to click and view all previous versions of that image.

2.4. Workflow State Management and Completion

The application shall ensure a robust, non-destructive, and stateful user workflow from start to finish.

- **State Persistence:** All user selections and generated content at each step of the workflow shall be automatically saved and preserved. This state must persist even if the user performs a full page refresh.
- **Rationale:** This requirement directly addresses the critical failure mode of context loss between sessions. The first prototype failed because it could not preserve state, leading to a disjointed and broken user experience. Full state persistence is non-negotiable.
- **Workflow Navigation:** The user shall be able to navigate backward to previous steps in the workflow (e.g., move from the "Thumbnails" step back to the "Hooks" step). When doing so, their progress and selections in all subsequent stages shall be retained.
- **Completion Screen:** Upon successfully finishing the final step of the workflow, the user shall be presented with a dedicated "Complete" screen. This screen will provide a comprehensive summary of all their final selections and generated assets from each step.These workflow specifications are enabled and enhanced by an integrated AI assistant that provides real-time, interactive support.

3.0 Integrated AI Assistant

3.1. Introduction to the AI Assistant

The integrated AI assistant is designed to be an active and capable participant in the content creation process, not merely a passive chatbot. It is empowered with the ability to understand contextual user requests and directly manipulate the application's user interface in real-time. This real-time interactivity overcomes the limitations of disconnected, backend-only systems and is a key differentiator that makes the assistant a true creative partner.

3.2. Core Capabilities

The AI assistant shall be equipped with the following functional capabilities to directly aid the user within the content generation workflow.

- **UI Placement:** The assistant's interface shall be presented in a dedicated panel, consistently available on the right-hand side of the content generation screens.
- **Real-Time UI Manipulation:** The assistant must be able to parse natural language commands and directly act upon the user's workspace, providing immediate visual feedback.
- **Editing Content:** The assistant shall process commands such as "please change card three from 'this' to 'that'" and have the text on the corresponding card in the UI update instantly. Upon successful modification, an "edited" badge shall appear on the card to signify the change.
- **Removing Content:** The assistant shall process commands like "Please remove card five", resulting in the corresponding card being immediately removed from the UI.
- **Rationale:** This capability is a primary differentiator. The first prototype's assistant could only manipulate data in a backend array, failing to update the UI. Direct,

real-time UI manipulation provides immediate visual feedback to the user, making the assistant a true interactive partner rather than a disconnected chatbot.

- **Content Versioning:** When the assistant edits a content card, a "History" button shall appear on that specific card. This function will allow the user to view all previous versions of the content, providing a clear audit trail of changes.
- **Proactive Suggestions:** The assistant shall provide contextual, clickable action buttons that suggest relevant commands to the user based on their current workflow stage (e.g., offering a "Regenerate All" button when viewing hooks).The combination of these features for core application management, a stateful content workflow, and an interactive AI assistant defines a highly capable, robust, and production-ready application for content creators.