SEOmation

Project Team

Syed Ibraheem 22i-2666
Abdul Rafay 22i-8785
Ibrahim Ahmed 22i-8781

Session 2022-2026

Supervised by

Dr. Asif Muhammad Malik

Co-Supervised by

None

Department of Computer Science

National University of Computer and Emerging Sciences
Islamabad, Pakistan

June, 2026

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement

Content creators, digital marketers, and small-to-medium enterprises face significant challenges in producing consistent, SEO-optimized content across multiple platforms due to the fragmented nature of existing tools and the prohibitive costs of comprehensive solutions. Current workflows require users to navigate between separate applications for keyword research, content generation, SEO optimization, image creation, platform-specific formatting, and publishing, creating inefficiencies and inconsistencies in content quality. Existing enterprise-level solutions such as SEMrush, Ahrefs, and HubSpot offer comprehensive features but are priced beyond the reach of individual creators and small businesses, with high monthly costs. Free and low-cost alternatives lack integration capabilities and fail to provide platform-specific content adaptation, forcing users to manually adjust content for different social media platforms and websites. The absence of real time SEO analysis tools that can provide actionable insights without requiring technical expertise further compounds the problem. Additionally, current AI content generation tools often produce generic outputs that lack SEO optimization, proper HTML structure, and multilingual support for global content strategies. This fragmentation results in sub optimal content performance, increased time investment, and reduced return on content marketing efforts, particularly affecting small businesses and independent creators who lack dedicated marketing teams or substantial budgets for premium tools.

SEOmation unifies these steps into a single platform that generates platform-ready content, performs basic on-page SEO checks, and streamlines scheduling/publishing.

1.2 Scope

SEOmation comprises:

- **Web App:** topic ideation; AI content generation (blogs, LinkedIn posts, Instagram posts — caption + image); multilingual support with initial focus on English; additional languages (e.g., Japanese, German, Russian) planned for future iterations; image generation with alt-text; editor with real-time SEO guidance; direct publishing & scheduling to WordPress, LinkedIn, and Instagram; calendar and status tracking.
- **Browser Extension:** on-page SEO audit; competitive comparison (URLs provided by user and/or current page); email extraction for convenience; PDF report export.

1.3 Modules

1.3.1 A) Web Application

1. **User Setup & Integration** — onboarding, profile/preferences, OAuth2 connections (WordPress, LinkedIn, Instagram), token management, language/platform settings.
2. **AI Topic & Content Generation** — topic suggestions; platform-specific drafts; RAG-assisted relevance; grammar/readability checks; HTML cleanup; multilingual (EN, JA, RU).
3. **Image Generation & Management** — AI image creation aligned with the content topic, alt-text generation, platform-specific sizes, style presets; associate images with posts across platforms (blogs/LinkedIn/Instagram). For Instagram: support auto-generated or manually uploaded images paired with captions, ready for publishing/scheduling.
4. **Editor & Review** — rich-text + HTML preview; platform preview; real-time SEO tips based on basic rules.
5. **Publishing & Scheduling** — publish now/later; calendar view; per-user timezone-aware scheduling; retries and status tracking; bulk actions; includes Instagram-optimized content preparation (caption + image); publishing via Instagram Graph API subject to approval, with manual export option as reliable alternative.

1.3.2 B) Browser Extension (SEO Analyzer)

6. **SEO Audit** — one-click checks: meta, headings, keyword density, images/alt-text, links; basic technical indicators.
7. **Competitive Analysis & Reporting** — side-by-side compare (selected URLs); rule-based score; gap/recommendations; PDF export.
8. **Email Extraction (Convenience)** — scrape visible emails from the current website for quick contact; respect site policies.

1.4 User Classes and Characteristics

Table 1.1: User Classes and Roles in the SEOmation System

User Class	Description
Creator	A solo creator or small-business marketer who uses the Web App for topic ideation, AI generation of blogs, LinkedIn posts, and Instagram posts (caption + image). They can generate content in supported language. They apply SEO guidance in the editor, schedule or publish content according to their local timezone, and use the Browser Extension to run on-page audits and extract emails from the current site.
Editor / Manager	Reviews content quality and SEO compliance in the Web App, requests competitor comparisons, and approves publishing schedules. Interacts with audit results and PDF reports for stakeholders. Requires clear visibility into status, errors, and suggested fixes.
Report Consumer (Client / Stakeholder)	Receives read-only outputs (audit and comparison PDFs or links) to review progress and provide approvals. Non-technical; needs concise, comprehensible summaries and rationale for recommendations. Typically does not log into the system UI.

Chapter 2

Project Requirements

2.1 Use Cases

In this section, we describe all the major use cases for the SEOmation system. Each use case is first summarized at a high level, and then expanded in detail.

2.1.1 Use Case 1: Generate Topic Ideas

//

USE CASE 1: Generate Topic Ideas	
Use Case	Generate Topic Ideas
Actors	Creator
Type	Primary
Description	The Creator requests topic ideas tailored to their business profile and current trends. The system returns a small batch of suggestions (e.g., 4–6).

//

EXPANDED USE CASE 1: Generate Topic Ideas	
Primary Actor	Creator
Goal	To provide creators with 4–6 targeted topic suggestions that align with their niche and current trends to jumpstart content development.
Stakeholders	Creator, System
Preconditions	(1) Creator logged in. (2) Business profile exists (or minimal prompt provided).
Postconditions	(1) Selected idea saved/attachable to a new draft (Blog/LinkedIn/Instagram).
Main Success Scenario	(1) User opens “Ideas”. (2) Selects business/industry + language. (3) (Optional) Adds keywords. (4) System returns 4–6 ideas. (5) User bookmarks or “Start Draft” from an idea. (6) System persists the selection.
Extensions	E1 “Regenerate” for a fresh batch. E2 No trend data available $\rightarrow$ fall back to profile-based ideas.

2.1.2 Use Case 2: Generate SEO Blog Article

//

USE CASE 2: Generate SEO Blog Article	
Use Case	Generate SEO Blog Article
Actors	Creator
Type	Primary
Description	The Creator generates an SEO-ready blog draft in any supported language. The system collects inputs (topic, keywords, language) and produces a structured draft with inline SEO guidance. The Creator may also generate or attach a featured image for the blog; the system stores the image with alt-text and links it to the draft.

//

EXPANDED USE CASE 2: Generate SEO Blog Article	
Primary Actor	Creator
Goal	To produce a draft SEO blog article with structured headings and SEO checks.
Stakeholders	Creator, Editor/Manager, System
Preconditions	(1) Creator has an active account. (2) User is logged in. (3) AI generation service is available.
Postconditions	(1) Draft saved as ContentItem. (2) SEO summary associated. (3) (Optional) Featured image associated with the draft.
Main Success Scenario	(1) User logs in. (2) Selects "New → Blog". (3) Provides topic/keywords/language. (4) System validates inputs. (5) AI generates draft with title/headings/meta/body. (6) SEO checks run. (7) Draft displayed with suggestions. (7a) (Optional) System suggests a featured image and alt-text or lets the user upload/select one. (8) User reviews/edits. (9) System saves draft.
Extensions	E1 API delay → progress shown. E2 Missing keywords → system suggests defaults. E3 Unsupported language → prompt correction.

2.1.3 Use Case 3: Generate LinkedIn Post

//

USE CASE 3: Generate LinkedIn Post	
Use Case	Generate LinkedIn Post
Actors	Creator
Type	Primary
Description	The Creator generates a LinkedIn post tailored to tone and platform length constraints. The Creator may attach or generate an accompanying image, resized for LinkedIn.

//

EXPANDED USE CASE 3: Generate LinkedIn Post	
Primary Actor	Creator
Goal	To produce concise LinkedIn content.
Stakeholders	Creator, System
Preconditions	(1) Creator logged in.
Postconditions	(1) Draft saved. (2) (Optional) Image asset stored and associated with the draft.
Main Success Scenario	(1) User selects "New → LinkedIn Post". (2) Provides goal, tone, language. (3) System generates 2–3 variants. (4) User selects/edits. (5) SEO/social heuristics applied. (5a) (Optional) System adds an image (generated or uploaded) and resizes it for LinkedIn. (6) Draft saved.
Extensions	E1 Regenerate variants. E2 Overlength warning.

2.1.4 Use Case 4: Generate Instagram Post

//

USE CASE 4: Generate Instagram Post	
Use Case	Generate Instagram Post
Actors	Creator
Type	Primary
Description	The Creator generates an Instagram caption and image for the same topic in any supported language. The image may be auto-generated or manually uploaded, and the post can be scheduled/published through Instagram APIs.

//

EXPANDED USE CASE 4: Generate Instagram Post	
Primary Actor	Creator
Goal	To prepare an Instagram post (caption + image) for scheduling/publishing.
Stakeholders	Creator, System
Preconditions	(1) Creator logged in. (2) Instagram account connected (for publish/schedule).
Postconditions	Caption and image stored and associated for publishing.
Main Success Scenario	(1) User selects "New → Instagram Post". (2) Provides prompt/topic, language, and optional hashtags. (3) System generates caption options. (4) System generates a relevant image (with alt-text) aligned to the topic; user may regenerate if desired. (5) User optionally uploads their own image instead. (6) System saves the caption + image pair as a draft.
Extensions	E1 No image accepted yet → keep caption draft, allow image later. E2 Generation error → retry or upload manually.
Special Requirements	Supports Instagram content preparation (photo + caption). Publishing via Instagram Graph API available for approved Business/Creator accounts; manual export supported as standard workflow.

2.1.5 Use Case 5: Edit Content with Real-Time SEO Suggestions

//

USE CASE 5: Edit Content with SEO Suggestions	
Use Case	Edit Content with SEO Suggestions
Actors	Creator, Editor/Manager
Type	Primary
Description	The user edits existing drafts with inline SEO hints provided by the system.

//

EXPANDED USE CASE 5: Edit Content with SEO Suggestions	
Primary Actor	Creator
Goal	Improve draft quality and SEO compliance.
Stakeholders	Creator, Editor/Manager, System
Preconditions	Draft exists.
Postconditions	Draft updated with SEO summary.
Main Success Scenario	(1) User opens draft. (2) SEO checks run. (3) Suggestions shown inline. (4) User edits. (5) System updates SEO score. (6) Draft saved.
Extensions	E1 Conflicting rules → explanation given. E2 Language change → checks updated.

2.1.6 Use Case 6: Connect Publishing Platforms via OAuth2

//

USE CASE 6: Connect Platforms	
Use Case	Connect Platforms
Actors	Creator
Type	Supporting
Description	The Creator links external platforms securely using OAuth2.

//

EXPANDED USE CASE 6: Connect Platforms	
Primary Actor	Creator
Goal	Enable publishing to linked platforms.
Stakeholders	Creator, External Platforms, System
Preconditions	User logged in.
Postconditions	Tokens stored securely.
Main Success Scenario	(1) User selects “Connect Platform”. (2) Redirected to OAuth consent. (3) User consents. (4) Tokens stored encrypted. (5) Connection verified.
Extensions	E1 Denied consent → instructions shown. E2 Expired token → refresh flow.
Special Requirements	Least privilege tokens; encryption at rest.

2.1.7 Use Case 7: Schedule and Publish Content

//

USE CASE 7: Schedule and Publish	
Use Case	Schedule & Publish
Actors	Creator, Editor/Manager
Type	Primary
Description	The Creator schedules or publishes drafts (blogs, LinkedIn posts, Instagram posts) to selected platforms with timezone support and retries.

//

EXPANDED USE CASE 7: Schedule and Publish	
Primary Actor	Creator
Goal	To publish content reliably at specified times.
Stakeholders	Creator, Editor/Manager, System, Target Platform
Preconditions	Draft exists; account connected; if the target platform supports or needs an image (e.g., Instagram photo, WordPress featured image, LinkedIn image post), an image is present.
Postconditions	Post published or marked failed; platform post IDs stored.
Main Success Scenario	(1) User selects draft. (2) Chooses platform(s) and timezone-aware time. (3) System validates fields (e.g., Instagram requires image). (4) System queues a job. (5) Worker publishes at run time (includes images where applicable—e.g., Instagram photo+caption; WordPress featured image; LinkedIn image share), then stores platform references. (6) System updates status and stores platform references.
Extensions	E1 Bulk scheduling. E2 Approval flow by Editor. E3 Platform-specific errors $\rightarrow$ user notified with remediation steps.
Special Requirements	DST handling; 3 retries with exponential backoff; monitoring hooks for failures. For non-Instagram platforms, image inclusion is subject to platform API capabilities (e.g., asset upload endpoints, content types).

2.1.8 Use Case 8: Run On-Page SEO Audit

//

USE CASE 8: Run SEO Audit	
Use Case	Run SEO Audit
Actors	Creator
Type	Primary
Description	The Creator runs a one-click audit of a webpage via the browser extension.

//

EXPANDED USE CASE 8: Run SEO Audit	
Primary Actor	Creator
Goal	To obtain a quick SEO report for a page.
Stakeholders	Creator, System
Preconditions	Extension installed.
Postconditions	Audit displayed; optional report saved.
Main Success Scenario	(1) User clicks “Audit Page”. (2) Extension collects metadata. (3) System computes score. (4) Results displayed.
Extensions	E1 Page blocks access $\rightarrow$ error shown.

2.1.9 Use Case 9: Compare Competitor Pages

USE CASE 9: Compare Competitors	
Use Case	Compare Competitors
Actors	Creator, Editor/Manager
Type	Primary
Description	User compares competitor content side-by-side with rule-based scores.

//

EXPANDED USE CASE 9: Compare Competitors	
Primary Actor	Creator
Goal	Identify content gaps against competitors.
Stakeholders	Creator, Editor/Manager, System
Preconditions	User provides URLs or uses current tab.
Postconditions	Comparison results saved.
Main Success Scenario	(1) User opens compare tool. (2) Enters competitor URLs. (3) System fetches pages. (4) Metrics computed. (5) Side-by-side view displayed. (6) Recommendations shown.
Extensions	E1 Blocked fetch → notify user.
Special Requirements	Respect robots.txt; attribute sources.

2.1.10 Use Case 10: Export Audit/Comparison as PDF

//

USE CASE 10: Export PDF	
Use Case	Export PDF
Actors	Creator, Editor/Manager, Report Consumer
Type	Supporting
Description	The user exports an audit or comparison as a PDF file for sharing.

//

EXPANDED USE CASE 10: Export PDF	
Primary Actor	Creator
Goal	To produce portable reports.
Stakeholders	Creator, Editor/Manager, Report Consumer
Preconditions	Audit/comparison exists.
Postconditions	PDF stored and downloadable.
Main Success Scenario	(1) User clicks export. (2) System renders PDF. (3) File saved with metadata. (4) User downloads/shares.
Extensions	E1 Error in rendering → retry.
Special Requirements	Accessible format; stable links.

2.1.11 Use Case 11: Extract Emails from Current Website

//

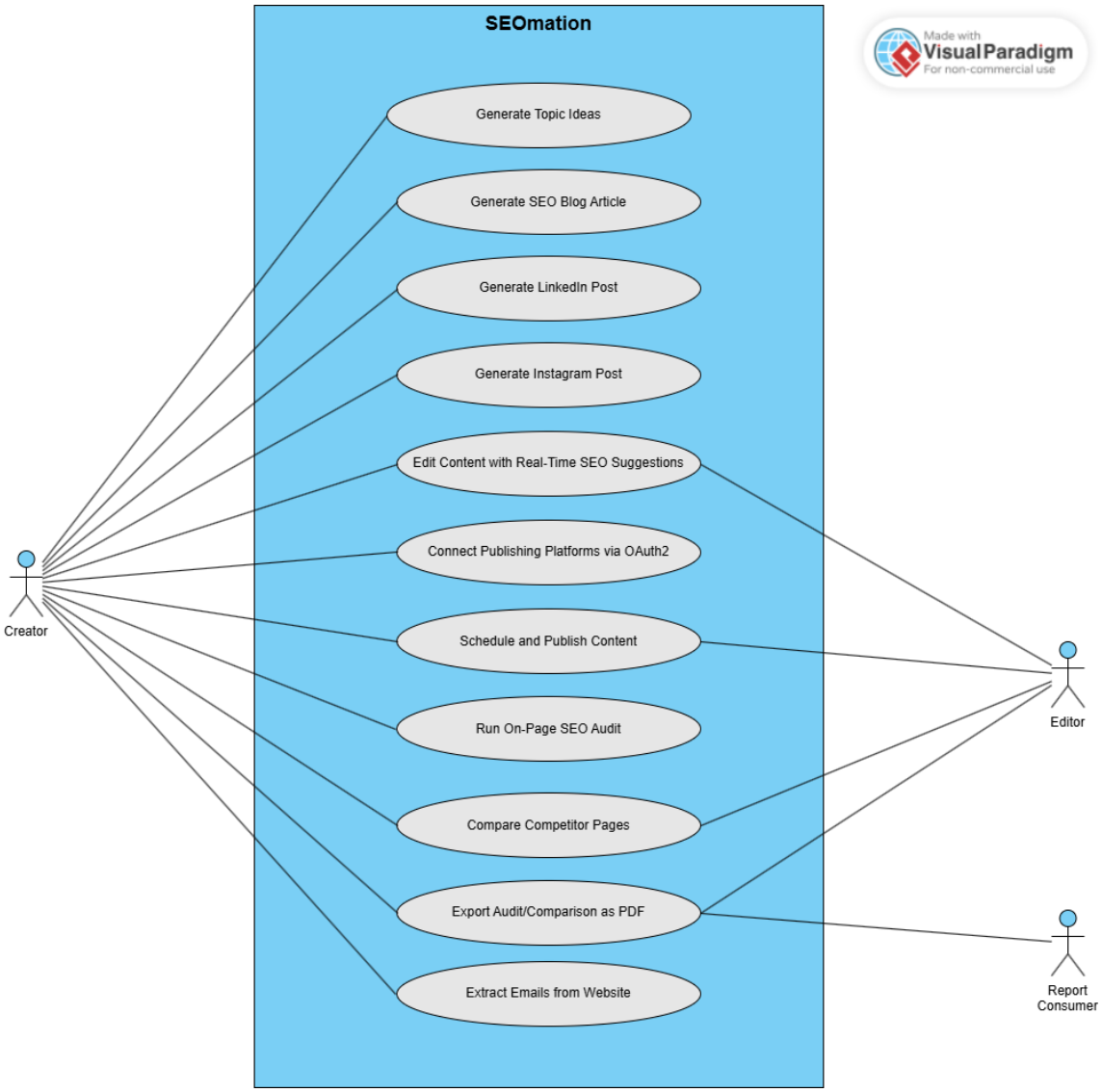
USE CASE 11: Extract Emails	
Use Case	Extract Emails
Actors	Creator
Type	Supporting
Description	The Creator extracts visible emails from the current site via extension.

//

EXPANDED USE CASE 11: Extract Emails	
Primary Actor	Creator
Goal	To capture email addresses for convenience.
Stakeholders	Creator, System
Preconditions	Extension active; site loaded.
Postconditions	Emails displayed; optional copy/export.
Main Success Scenario	(1) User clicks extract. (2) Extension scans DOM. (3) Results deduplicated and shown.
Extensions	E1 No emails found $\rightarrow$ display message.
Special Requirements	Privacy disclaimer shown; no storage by default.

2.2 Use Case Diagram

Below is the Use Case Diagram of the system:



2.3 Functional Requirements

This section specifies the Functional Requirements (FRs) of the SEOmation system in a structured, testable, and unambiguous manner. Requirements are grouped by modules corresponding to the system's architecture (Web App and Browser Extension). Each requirement is uniquely numbered for traceability.

2.3.1 Module A1 — User Setup & Integration

- **FR-A1-1:** The system shall allow users to register, authenticate, and manage their accounts securely.
- **FR-A1-2:** The system shall allow users to create and update a profile, specifying

business details, goals, and preferred language (initially English; additional languages in future releases).

- **FR-A1-3:** The system shall support OAuth2 connections to WordPress, LinkedIn, and Instagram, with secure token storage and automatic refresh.
- **FR-A1-4:** The system shall verify each platform connection and report its status to the user.
- **FR-A1-5:** The system shall allow users to disconnect an integration at any time.

2.3.2 Module A2 — AI Topic & Content Generation

- **FR-A2-1:** The system shall generate a batch of 4–6 topic ideas per request, tailored to the user's business profile, selected language, and optional competitor references.
- **FR-A2-1a:** The system shall support "Regenerate" to fetch a new batch without losing the previous batch.
- **FR-A2-1b:** The system shall allow the user to bookmark ideas and start a new draft (Blog/LinkedIn/Instagram) directly from a selected idea.
- **FR-A2-2:** The system shall generate platform-specific drafts (blogs, LinkedIn posts, Instagram posts) in any supported language, based on user inputs.
- **FR-A2-3:** The system shall allow users to specify a topic, keywords, and optional competitor URLs or references to guide generation.
- **FR-A2-4:** The system shall provide multiple draft variants where relevant (e.g., 2–3 LinkedIn post versions).
- **FR-A2-5:** The system shall structure generated blogs with headings, meta description suggestions, and clean HTML formatting.
- **FR-A2-6:** The system shall apply grammar, readability, and style checks to all drafts.
- **FR-A2-7:** The system shall save all generated drafts persistently, linked to the user's account.

2.3.3 Module A3 — Image Generation & Management

- **FR-A3-1:** The system shall generate images relevant to the user's topic and associate them with drafts and posts.

- **FR-A3-2:** The system shall generate and suggest descriptive alt-text for images.
- **FR-A3-3:** The system shall provide platform-specific image sizes and simple style presets.
- **FR-A3-4:** The system shall allow either auto-generated images or manual image upload/selection by the user (without making manual selection mandatory for any platform).
- **FR-A3-5:** The system shall store image assets with metadata for later publishing or reuse.

2.3.4 Module A4 — Editor & Review

- **FR-A4-1:** The system shall provide a WYSIWYG editor with HTML preview capability.
- **FR-A4-2:** The system shall display real-time SEO suggestions (e.g., title length, H1 presence, keyword usage, alt-text coverage) while editing.
- **FR-A4-3:** The system shall allow users to accept, ignore, or override SEO suggestions.
- **FR-A4-4:** The system shall support multilingual editing and preview in the chosen target language.

2.3.5 Module A5 — Publishing & Scheduling

- **FR-A5-1:** The system shall allow immediate publishing of content to connected platforms, including Instagram photo + caption publishing via Instagram Graph API (subject to platform policies).
- **FR-A5-2:** The system shall allow scheduling of content publication at user-specified times in their local timezone, with DST adjustments.
- **FR-A5-3:** The system shall queue scheduled jobs and execute them reliably using a background worker.
- **FR-A5-4:** The system shall track and display publishing status (Scheduled, Published, Failed).
- **FR-A5-5:** The system shall retry failed publishing operations up to three times with exponential backoff.
- **FR-A5-6:** The system shall allow bulk publishing and scheduling of multiple drafts.

2.3.6 Module B6 — SEO Audit (Browser Extension)

- **FR-B6-1:** The extension shall allow users to trigger an SEO audit on the currently loaded webpage with one click.
- **FR-B6-2:** The system shall extract metadata, headings, keyword density, image alt-text presence, and internal/external links from the page.
- **FR-B6-3:** The system shall compute a rule-based SEO score for the audited page.
- **FR-B6-4:** The system shall display prioritized recommendations for improvement.
- **FR-B6-5:** The system shall allow users to save audit results and associate them with their account.

2.3.7 Module B7 — Competitive Analysis & Reporting

- **FR-B7-1:** The system shall accept user-provided competitor URLs and/or use the current tab for comparison.
- **FR-B7-2:** The system shall fetch competitor content in compliance with robots.txt and rate-limit policies.
- **FR-B7-3:** The system shall analyze competitor pages and compute metrics according to the same SEO rules applied to user drafts.
- **FR-B7-4:** The system shall display side-by-side comparisons, highlighting gaps and strengths.
- **FR-B7-5:** The system shall allow exporting comparison results as PDF reports.

2.3.8 Module B8 — Email Extraction

- **FR-B8-1:** The extension shall scan the visible content of the current webpage to identify email addresses.
- **FR-B8-2:** The system shall display extracted email addresses to the user in a deduplicated list.
- **FR-B8-3:** The system shall allow users to copy or export the extracted email addresses.
- **FR-B8-4:** The system shall not persist email data by default unless the user explicitly saves it.

2.4 Non-Functional Requirements

- **NFR-1 (Performance):** The system shall generate short-form content (LinkedIn/Instagram posts) within 10 seconds and long-form content (blog articles) within 60 seconds under normal network conditions for a single user. Response times may vary based on external API availability.

- **NFR-2 (Reliability):** Scheduled content shall be published at the user-specified time (within 5 minutes). Failed publishing attempts shall be retried up to 3 times. Scheduled jobs shall persist in the queue system and survive application restarts.

- **NFR-3 (Security & Privacy):** OAuth tokens shall be encrypted at rest and transmitted over TLS; platform scopes shall follow a least-privilege policy; email extraction results are not stored by default and require explicit user action to save; competitor fetching shall respect robots.txt and rate limits.

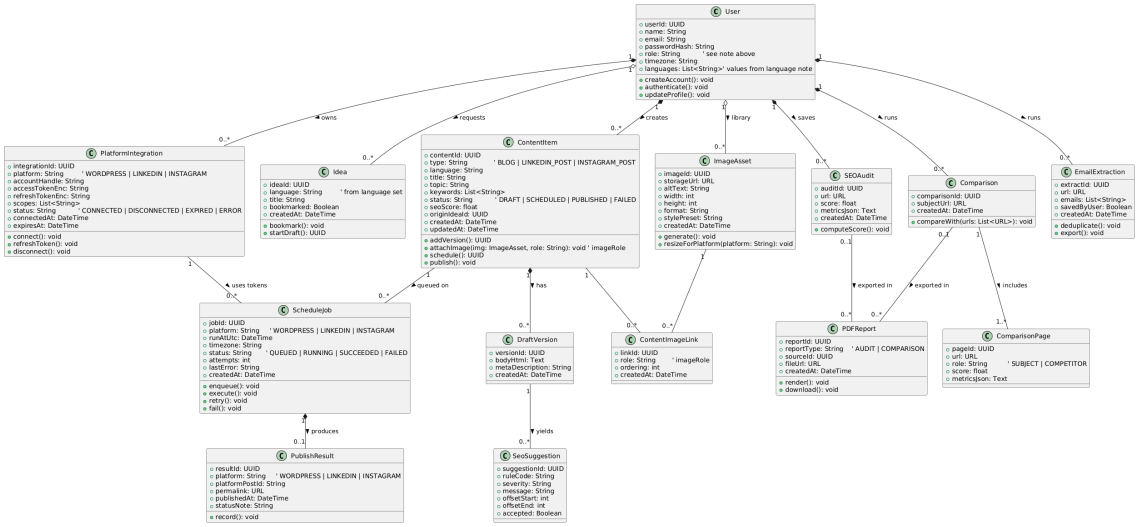
- **NFR-4 (Usability):** Editor SEO hints shall appear within 1 second after the user stops typing; the extension shall provide a one-click audit action; publish/schedule flows shall display clear success/failure messages with actionable next steps.

- **NFR-5 (Compatibility):** The web application shall function correctly in the latest versions of Chrome, Edge, and Firefox browsers. The browser extension shall support Chromium-based browsers using Manifest V3.

- **NFR-6 (Maintainability & Modularity):** The system shall be designed with modular components (separate services for AI generation, SEO audit, publishing, and extension) to simplify debugging and allow independent updates; source code shall follow standard documentation and naming conventions.

2.5 Domain Model

Below is the Domain Model of the project:


User
owns
requests
creates
library
saves
runs
runs
PlatformIntegration
Idea
ContentItem
ImageAsset
SEOAudit
Comparison
EmailExtraction
uses tokens
queued on
has
exported in
exported in
includes
ScheduleJob
DraftVersion
ContentImageLink
PDFReport
ComparisonPage
produces
yields
PublishResult
SeoSuggestion

Chapter 3

System Overview

This chapter provides a general description of the functionality, context, and design of the SEOmation system. It also illustrates the architecture and design models used.

3.1 Architectural Design

3.1.1 Simple Architecture

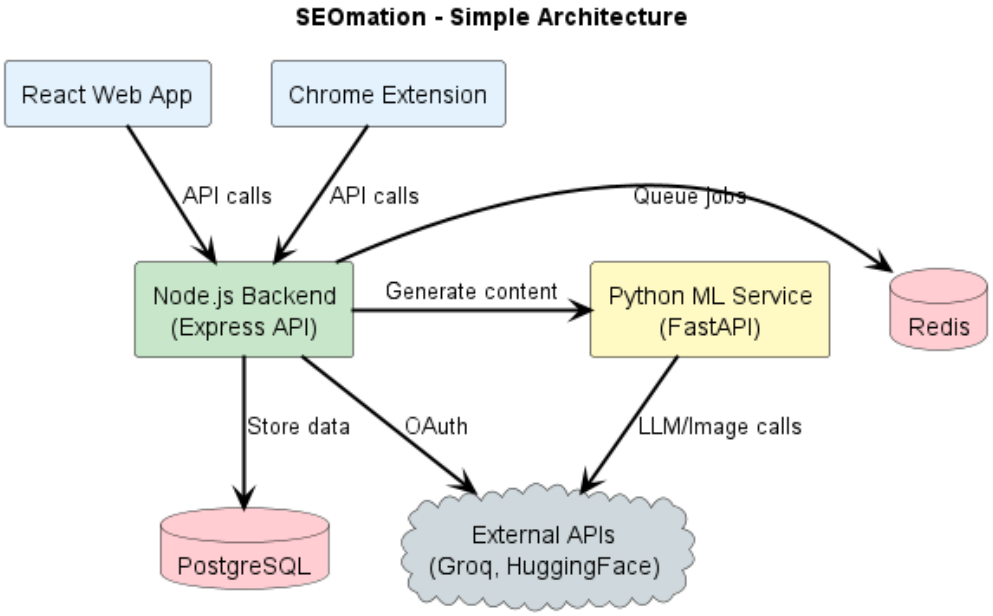


Figure 3.1: Simple Architecture

3.1.2 Detailed Architecture

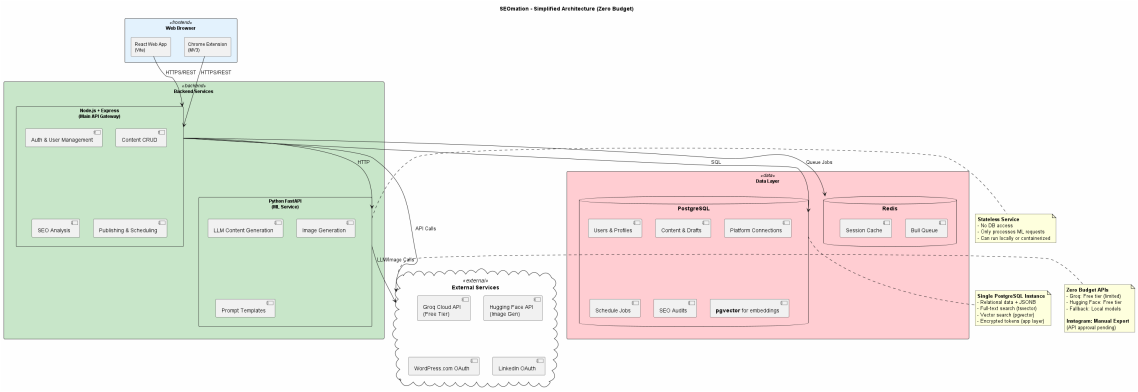


Figure 3.2: Detailed Architecture

3.1.3 Very Detailed Architecture

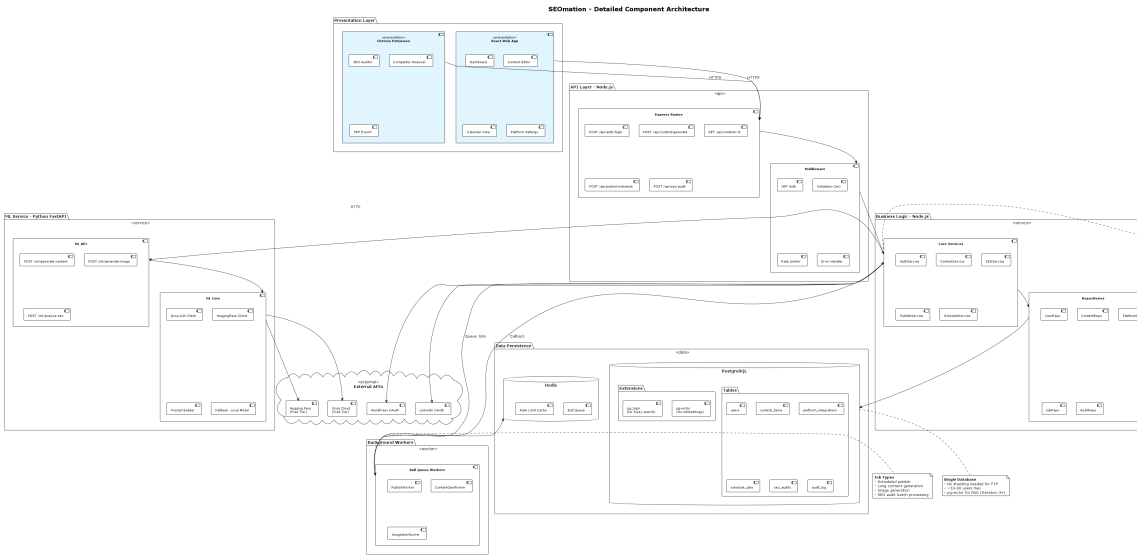


Figure 3.3: Very Detailed Architecture

3.2 Design Model

This section presents the detailed design models of the SEOmation system.

3.2.1 Activity Diagram

3.2.1.1 Activity Diagram — Web Application

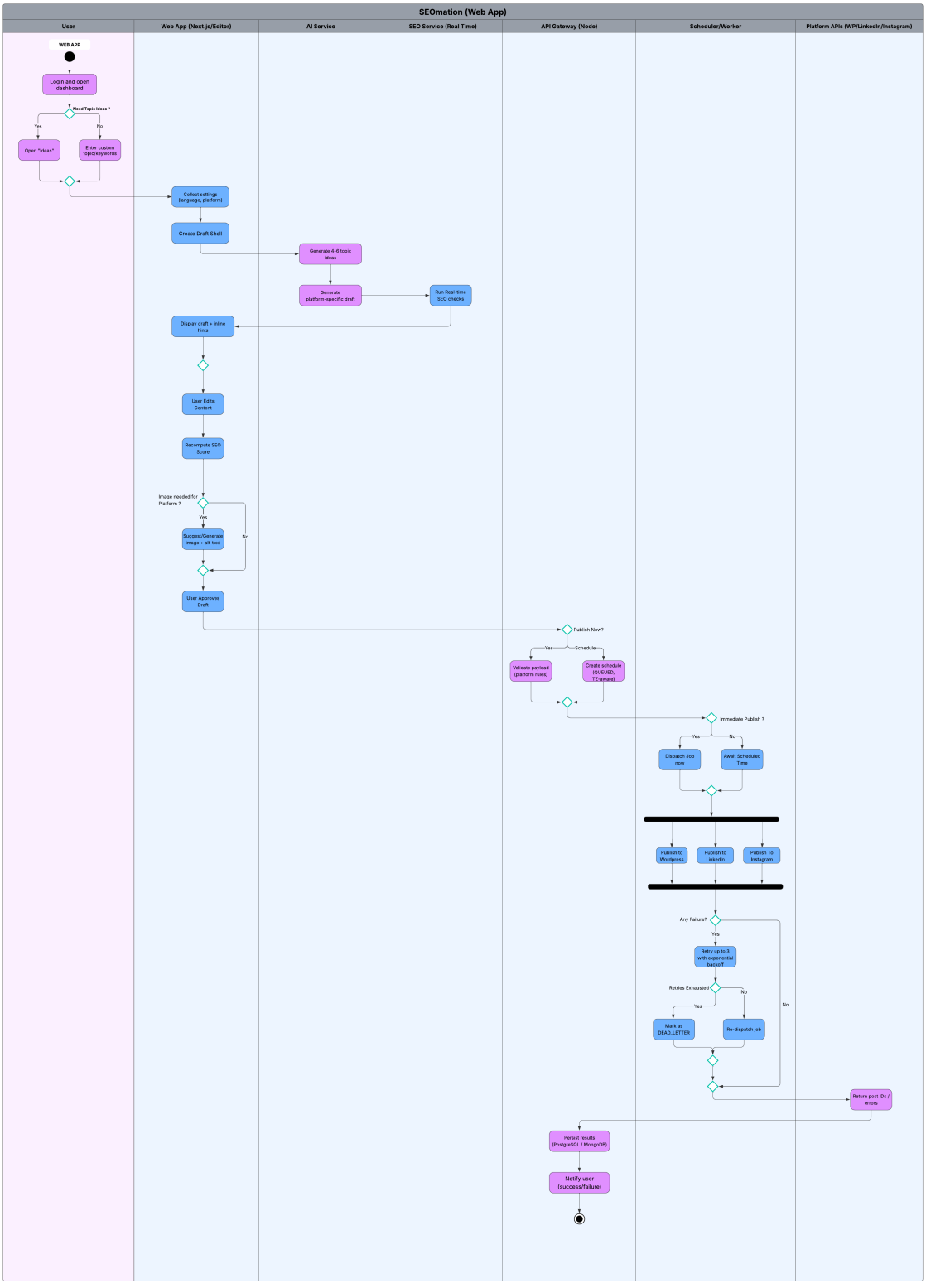


Figure 3.4: Activity Diagram — Web Application

3.2.1.2 Activity Diagram — Extension

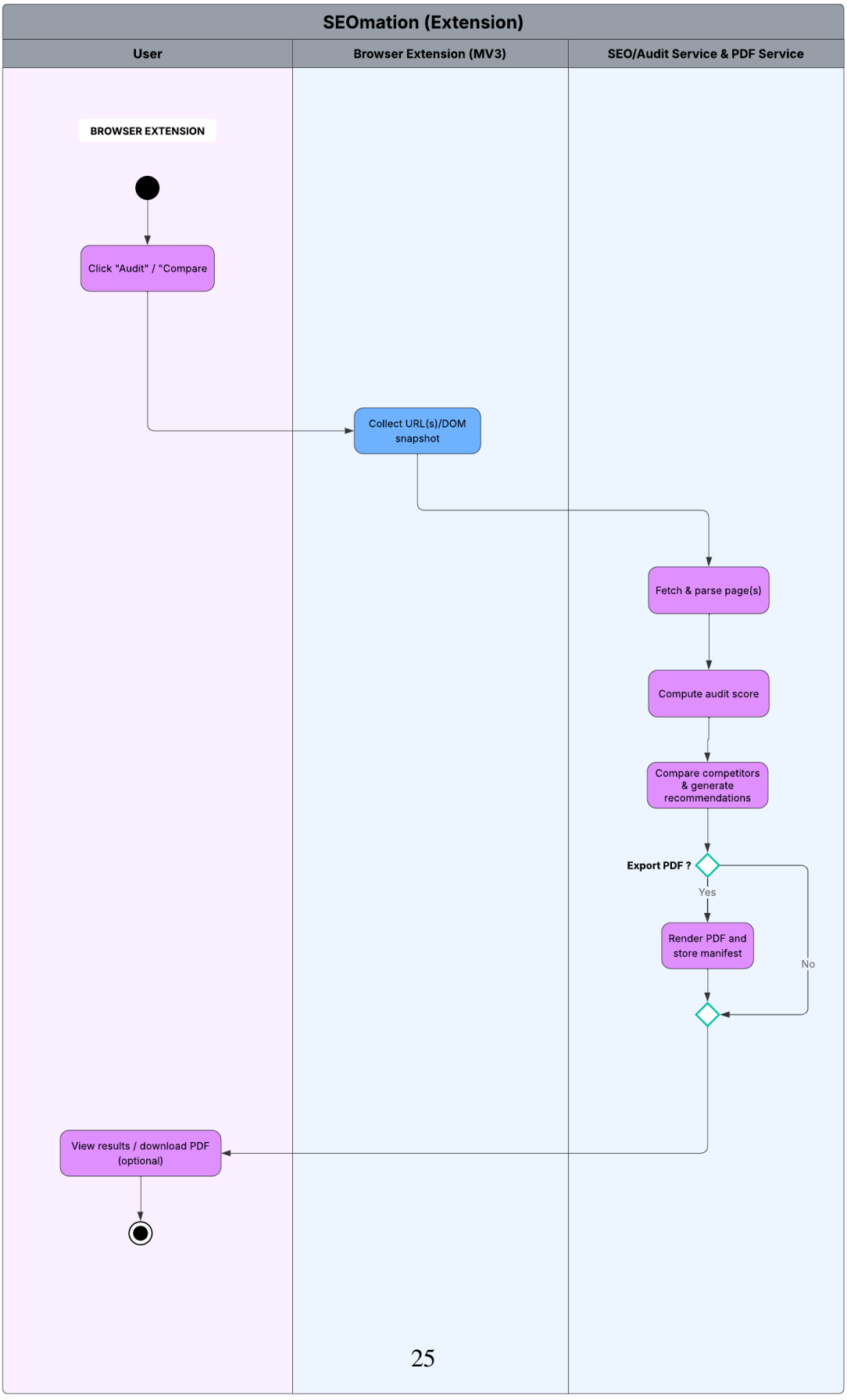


Figure 3.5: Activity Diagram — Extension

3.2.2 Data Flow Diagrams

3.2.2.1 Level 0

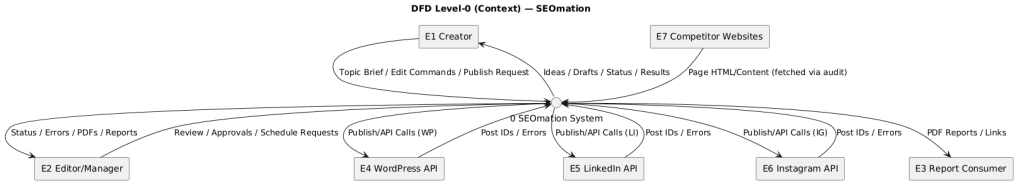


Figure 3.6: Data Flow Diagram — Level 0

3.2.2.2 Level 1

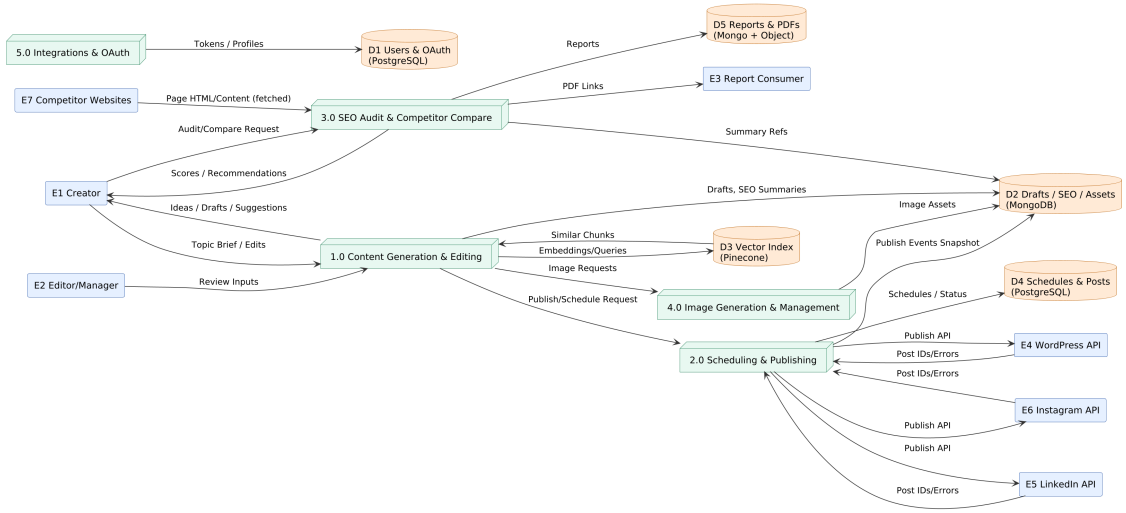


Figure 3.7: Data Flow Diagram — Level 1

3.2.2.3 Level 2.1

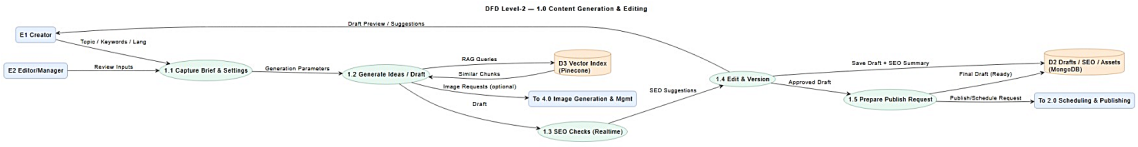


Figure 3.8: Data Flow Diagram — Level 2.1

3.2.2.4 Level 2.2

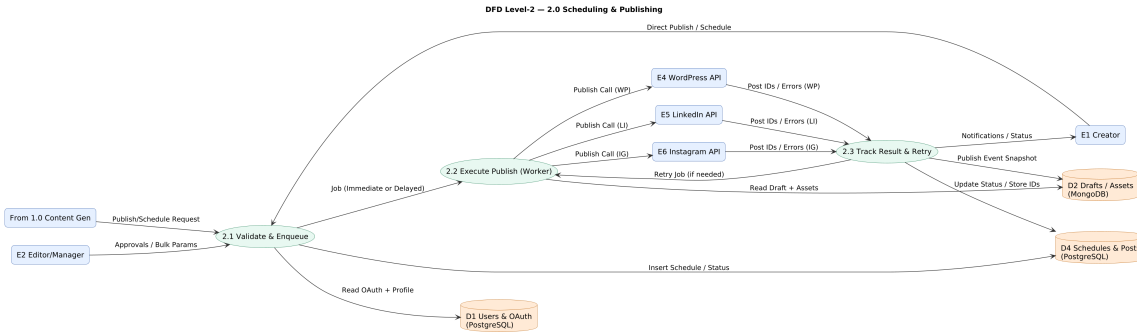


Figure 3.9: Data Flow Diagram — Level 2.2

3.2.2.5 Level 2.3

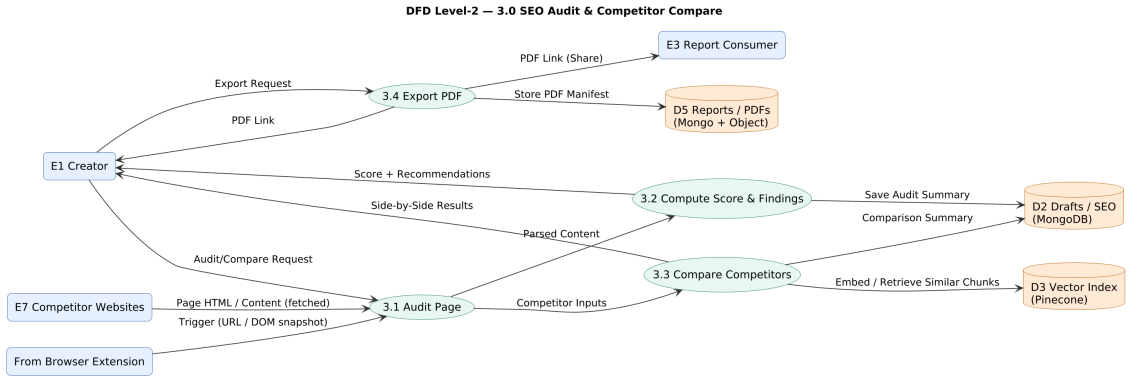


Figure 3.10: Data Flow Diagram — Level 2.3

3.2.2.6 Level 2.4

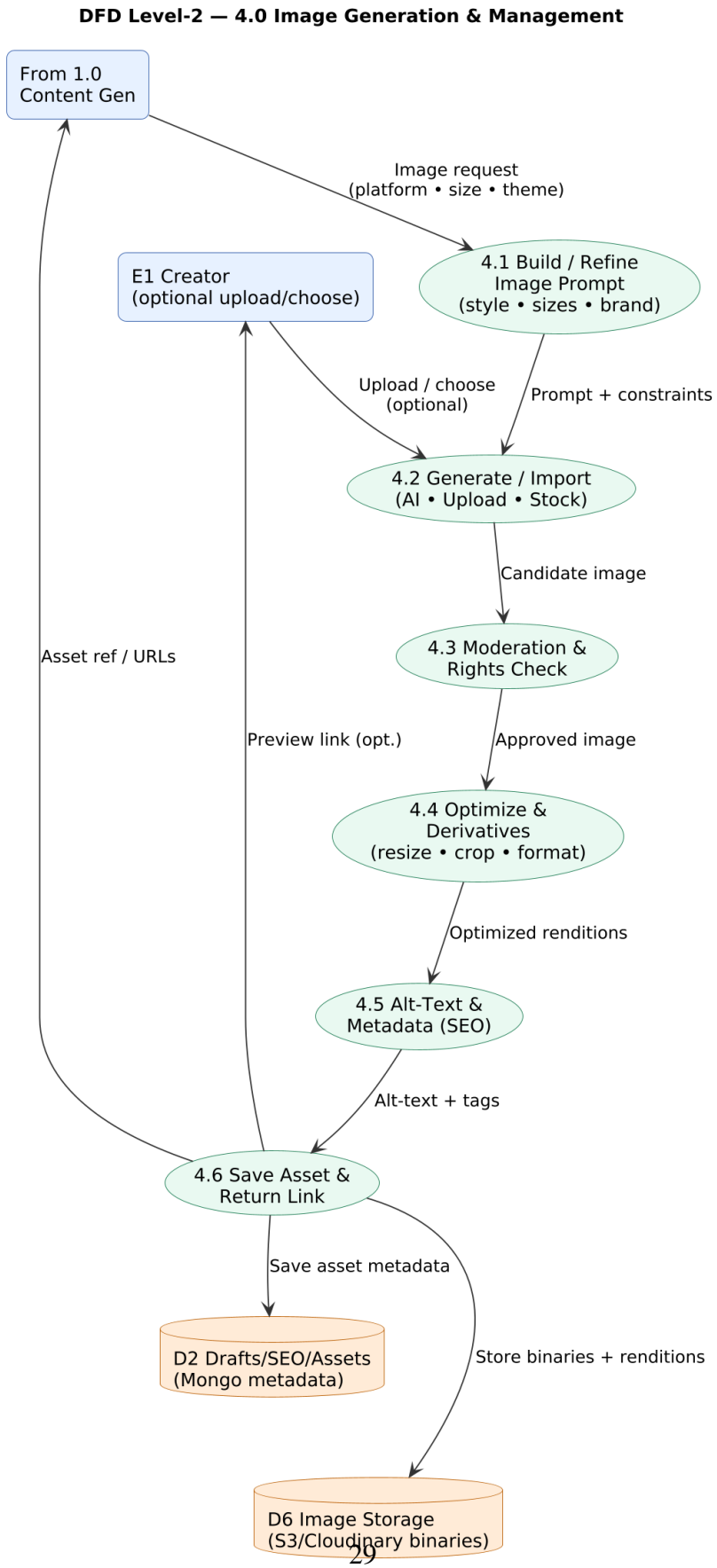


Figure 3.11: Data Flow Diagram — Level 2.4

3.2.2.7 Level 2.5

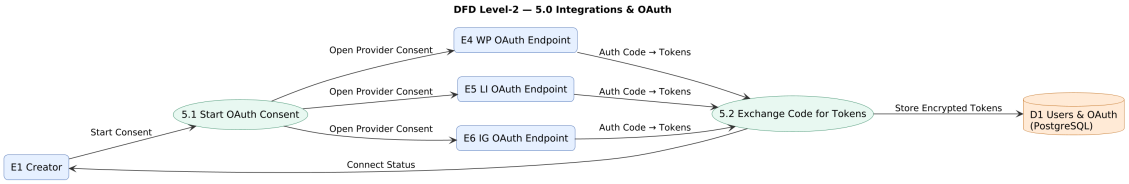


Figure 3.12: Data Flow Diagram — Level 2.5

3.2.3 System-level Sequence Diagrams

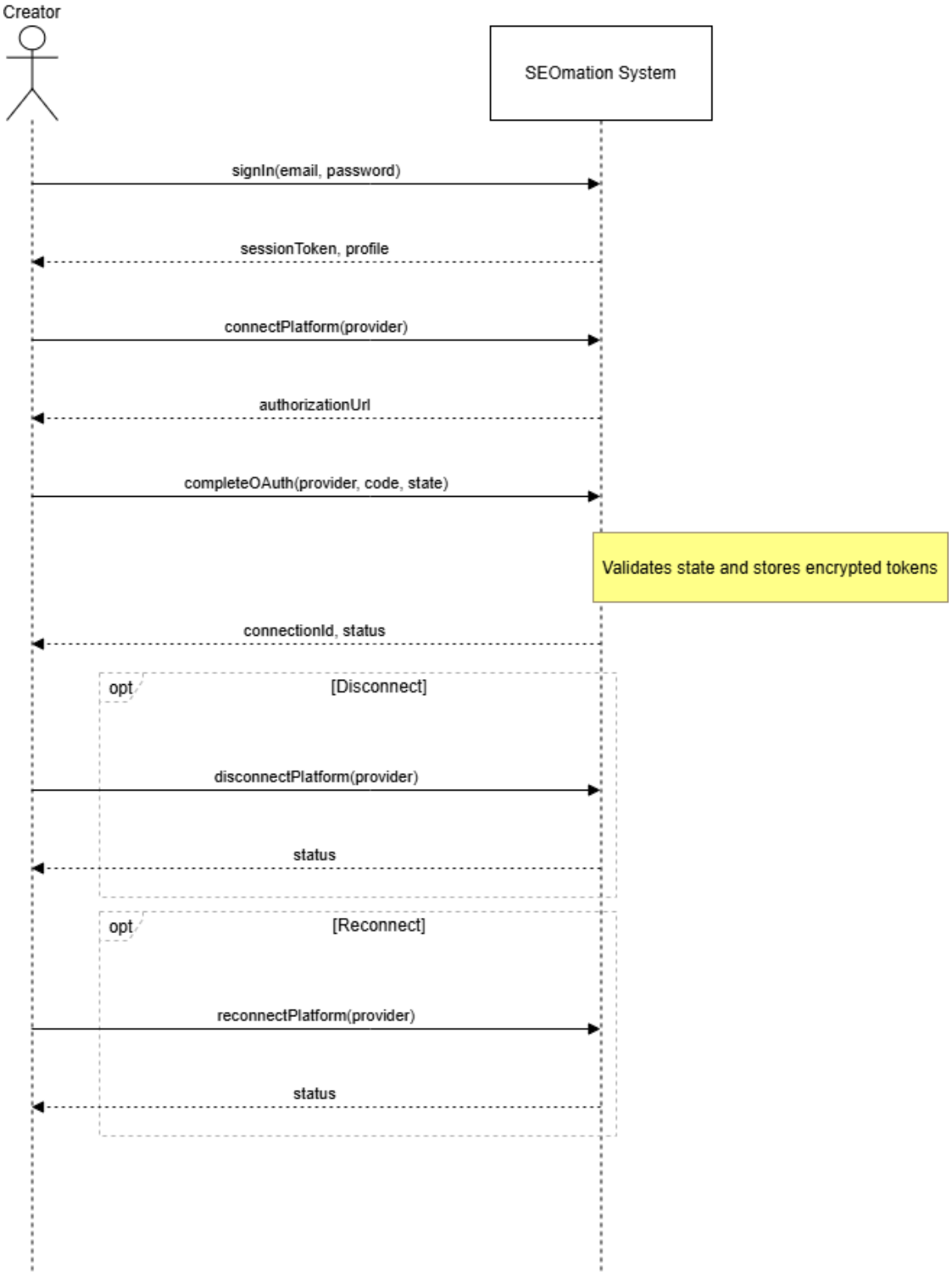


Figure 3.13: SSD-1 — Account Access & Platform Connection

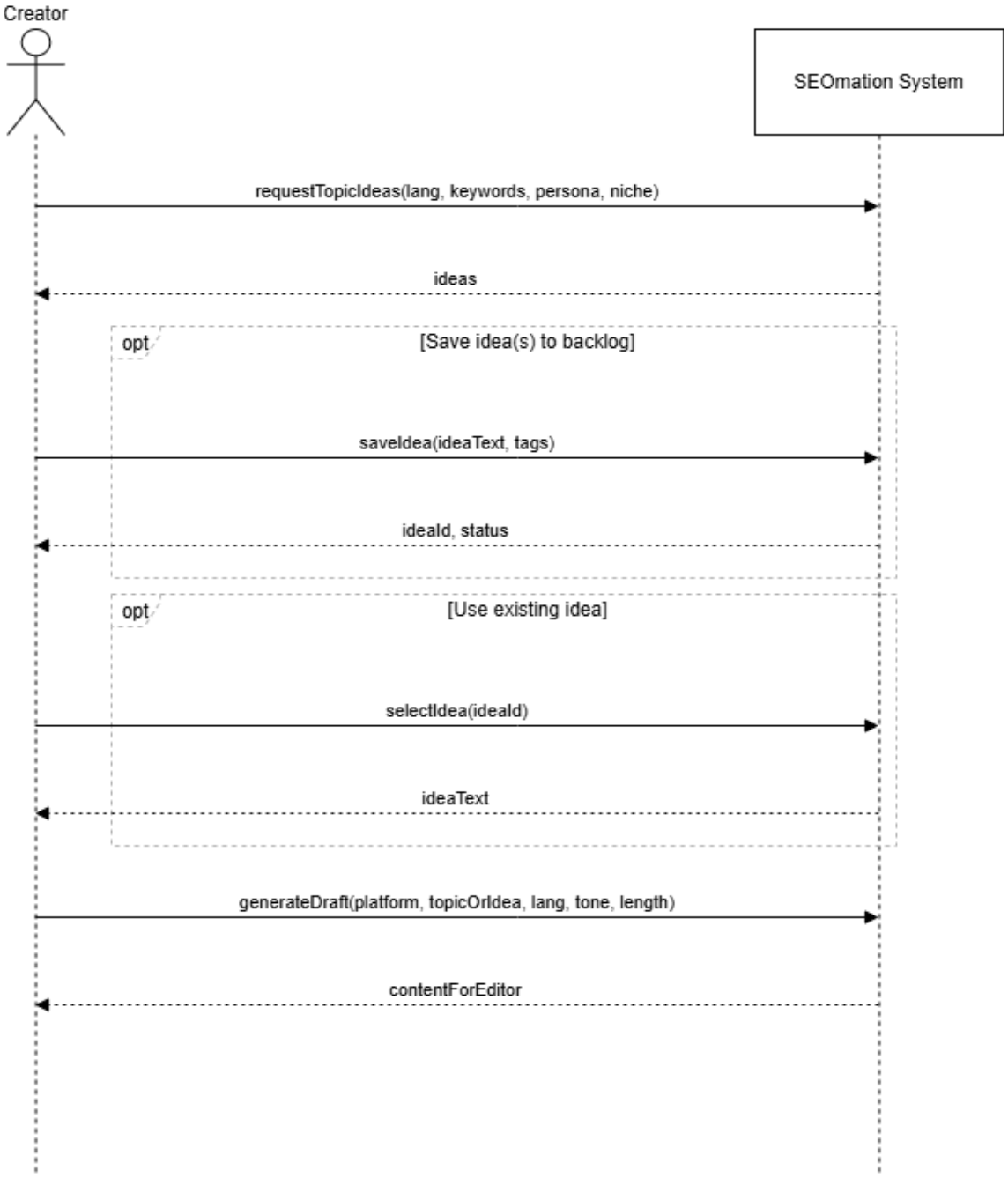


Figure 3.14: SSD-2 — Ideation & Idea Drafts Generate Content

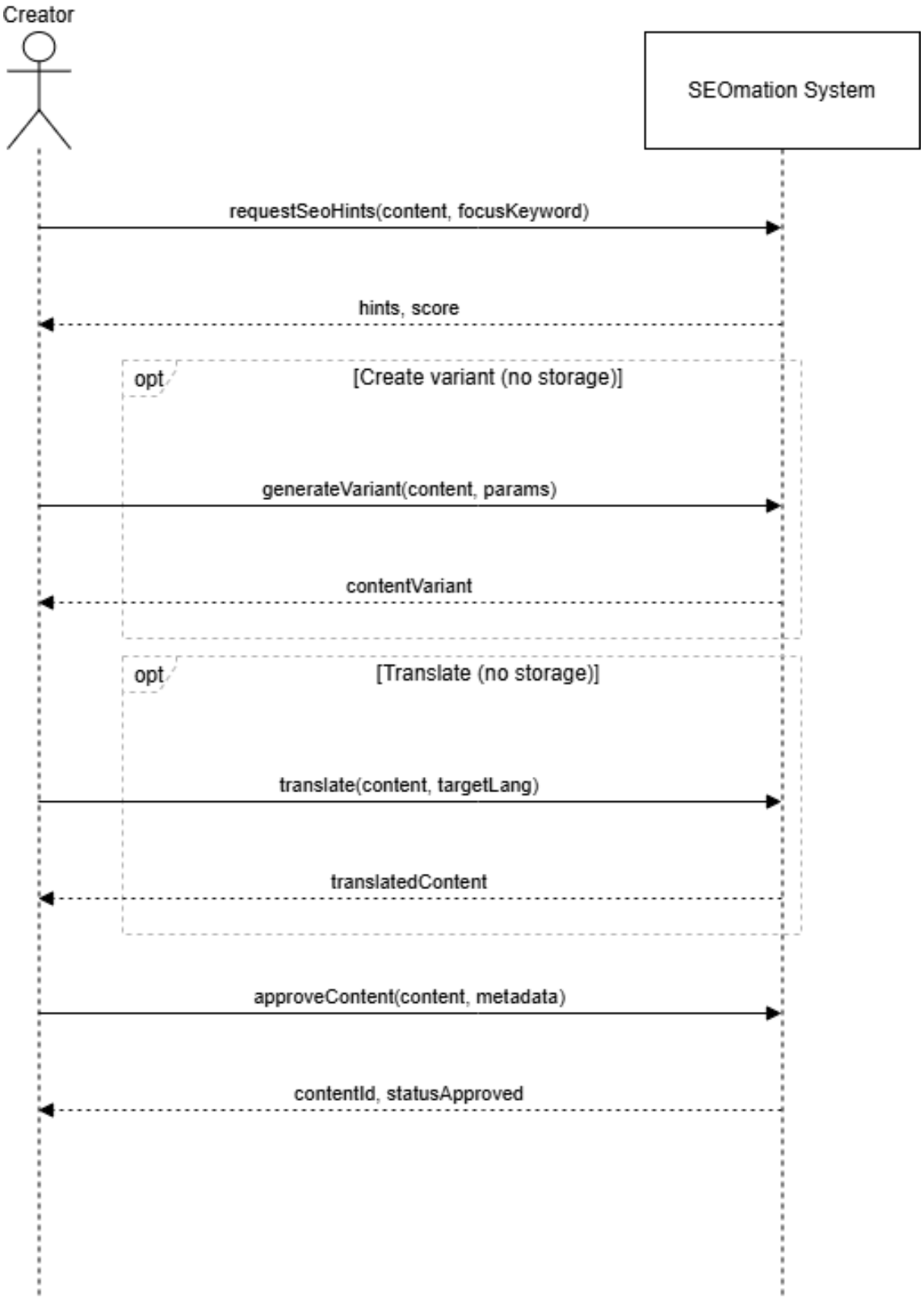


Figure 3.15: SSD-3 — Iterate Live, SEO Hints, Approve Final Content

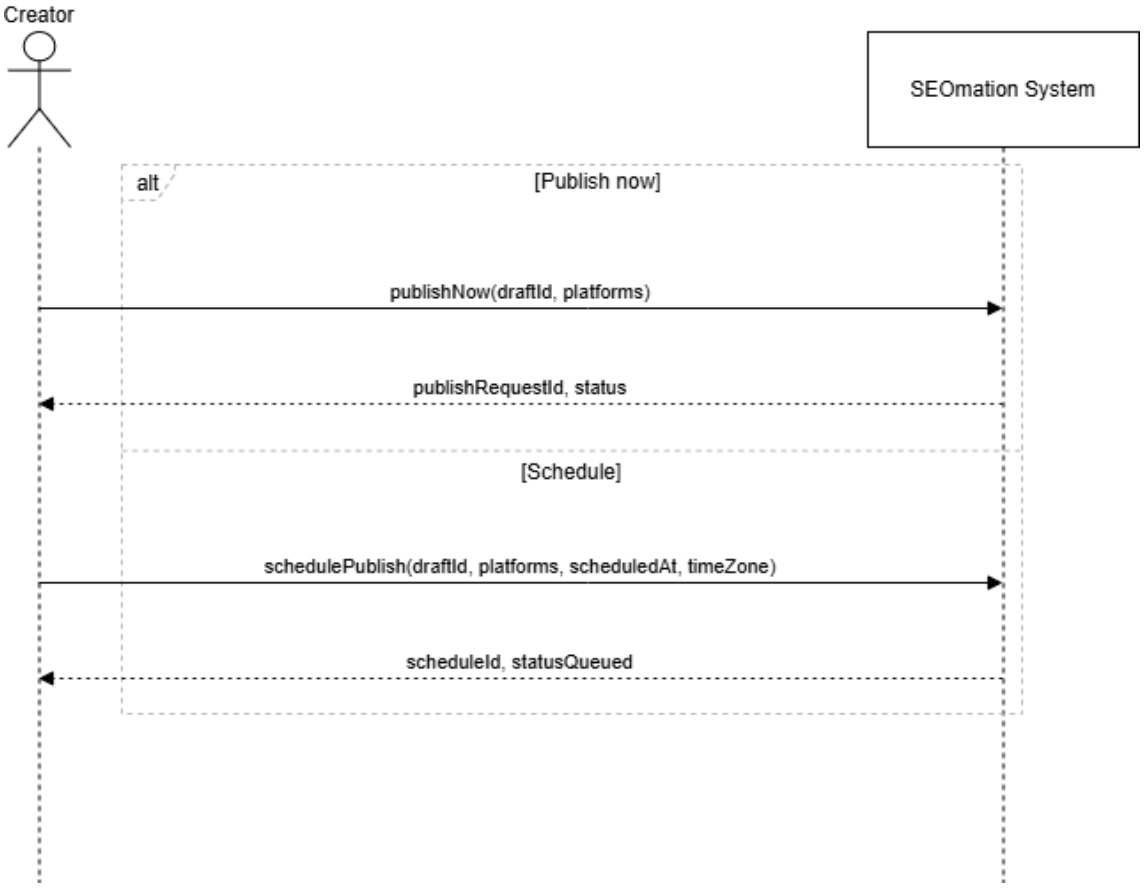


Figure 3.16: SSD-4 — Publish Now or Schedule

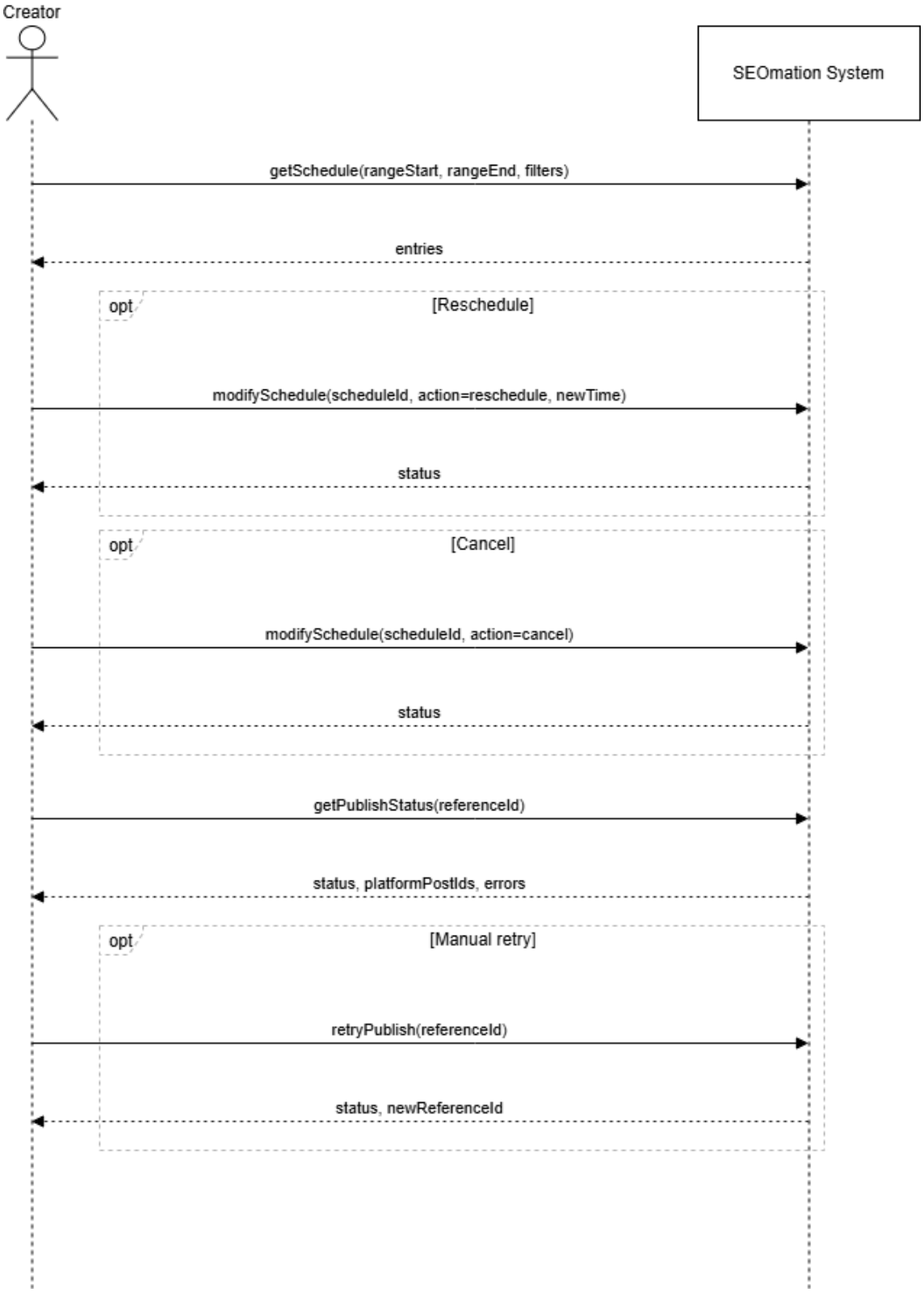


Figure 3.17: SSD-5 — Manage & Monitor Publishing

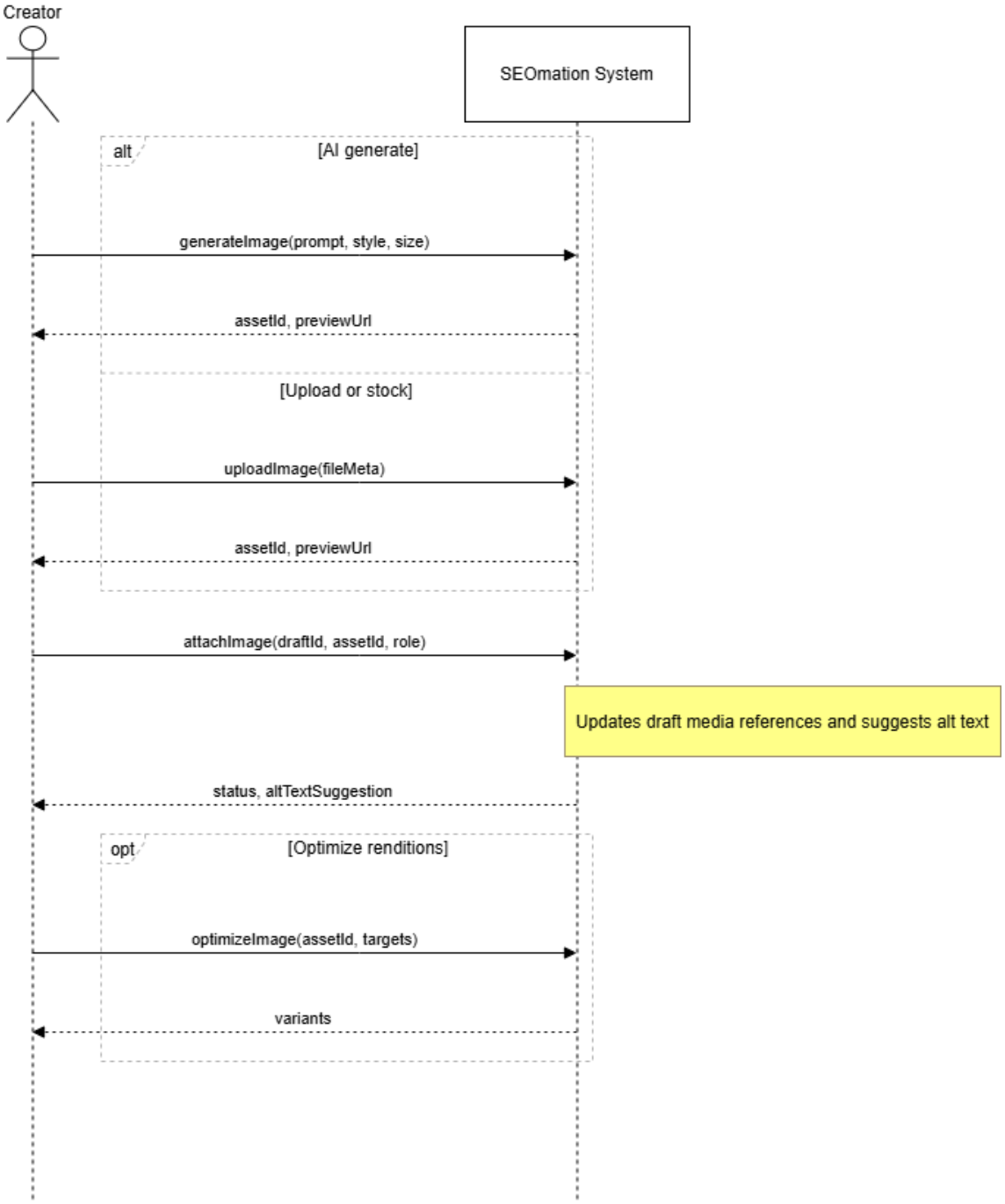


Figure 3.18: SSD-6 — Image Pipeline: Generate, Upload, Attach, Optimize

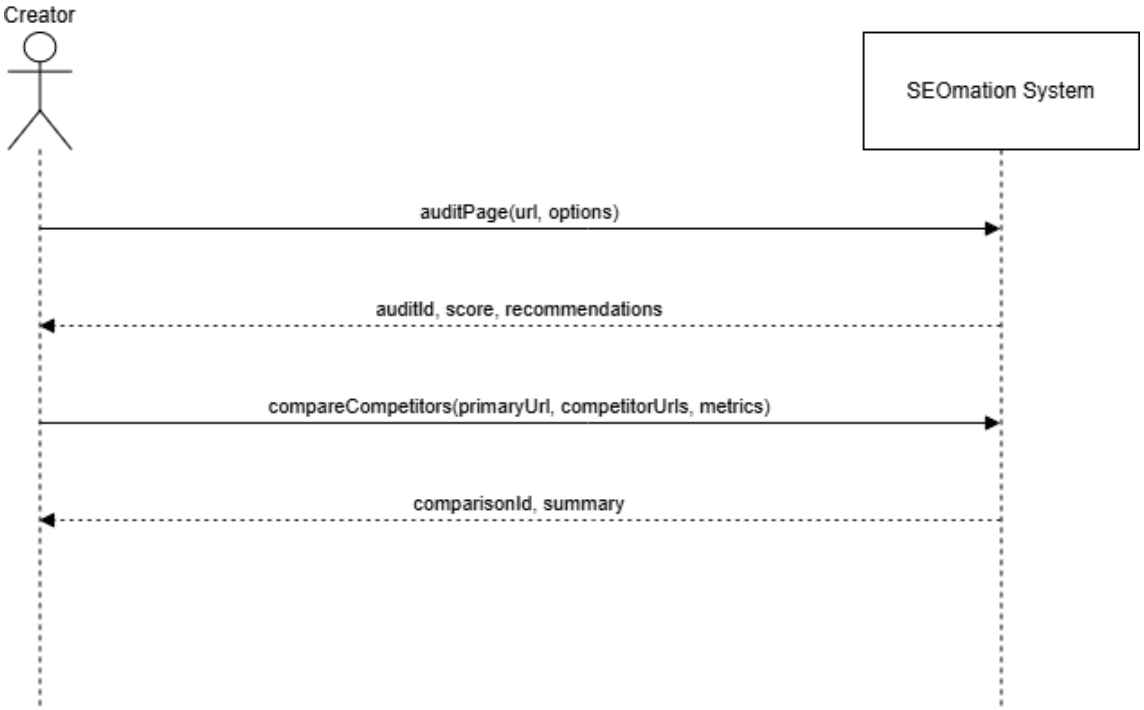


Figure 3.19: SSD-7 — Audit & Compare (Extension)

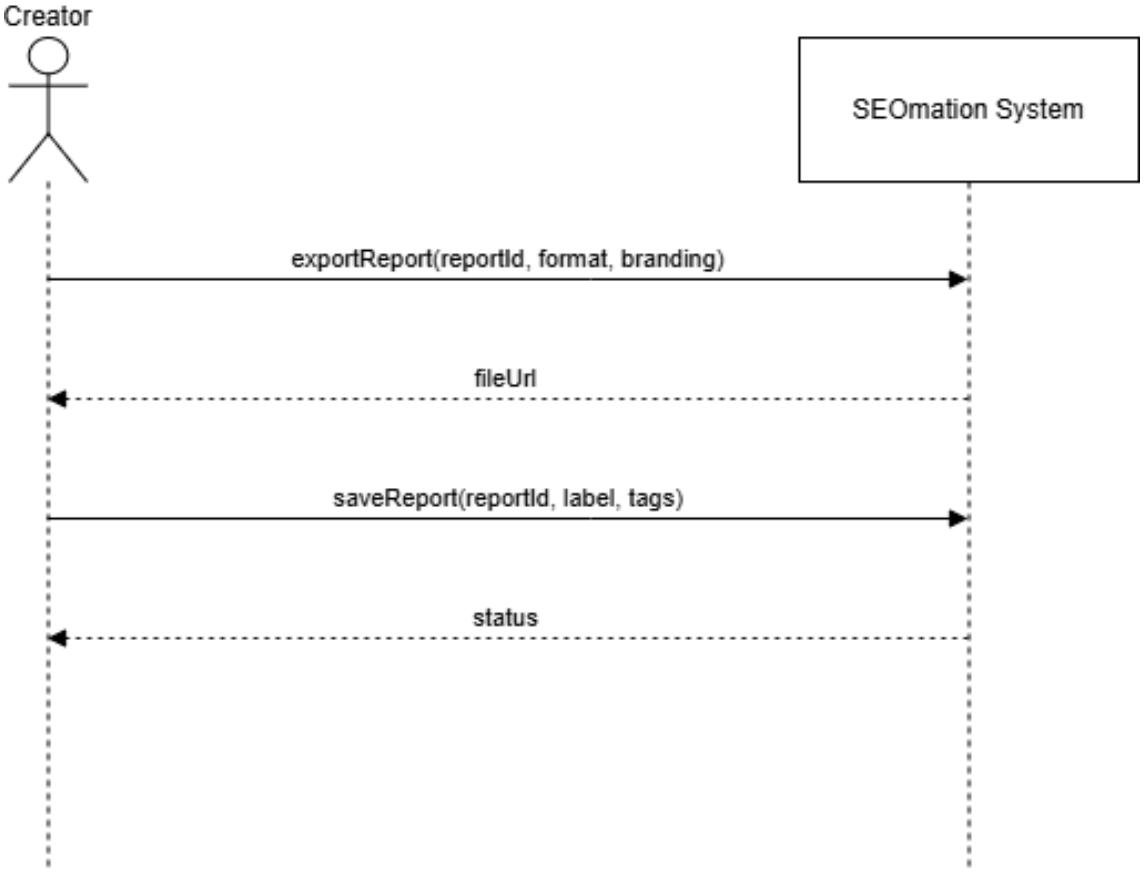


Figure 3.20: SSD-8 — Export & Save Report (Extension)

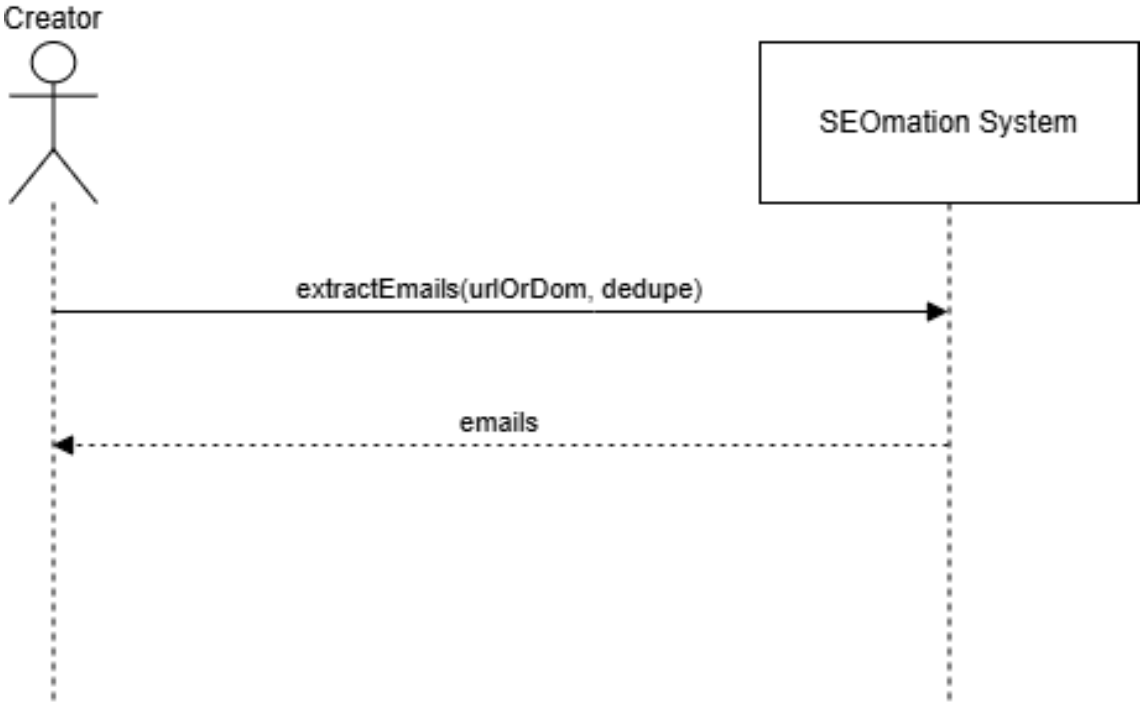


Figure 3.21: SSD-9 — Extract Emails (Extension)

3.3 Data Design

Storage Choice (Summary)

We use a relational store (**PostgreSQL 15+**) for strong consistency, referential integrity, and mature indexing. Core data types include UUID primary keys, enums for finite states (e.g., content type/status, platform), and JSONB for small, flexible metadata where needed. This aligns with our modular architecture and supports clean joins for the most common read paths (draft → schedule → publish).

Conceptual/Logical Model (ER)

Figure 3.22 presents a compact ER view of SEOmation's core data. We intentionally limit the diagram to eight entities to keep relationships and keys clear for evaluation.

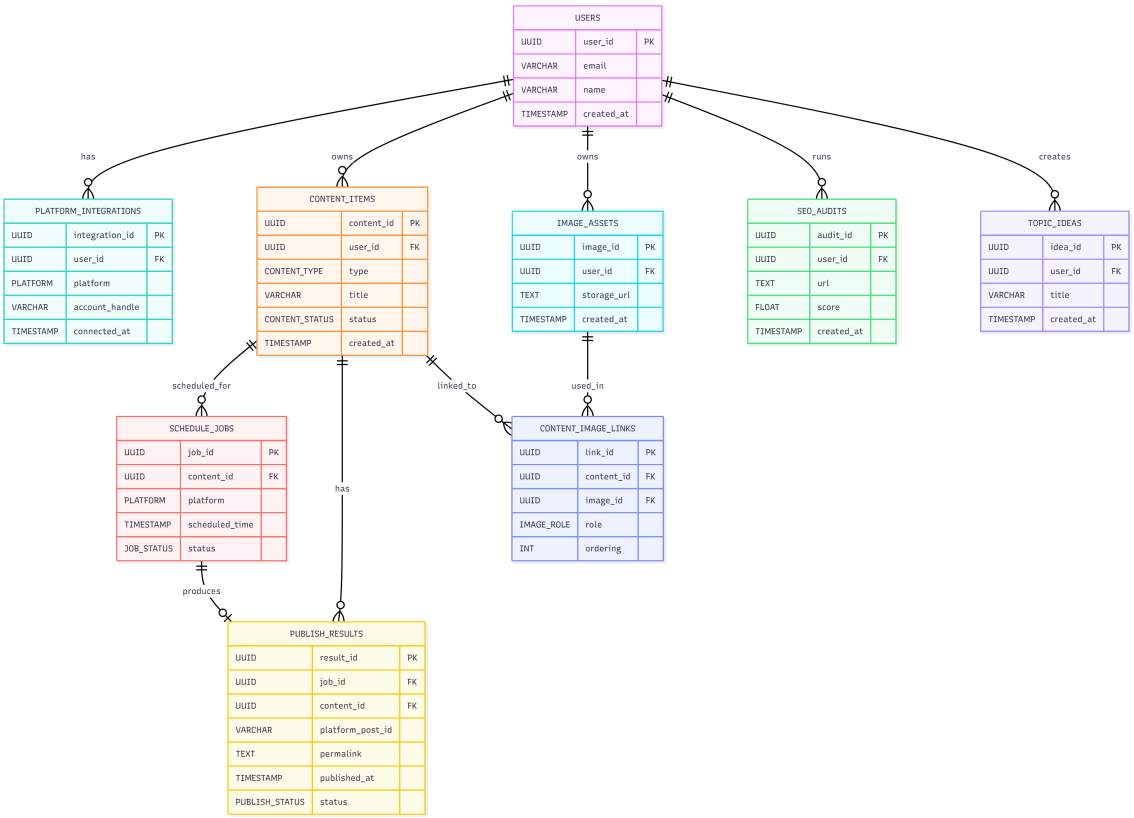


Figure 3.22: SEOmation ER (simplified): core entities, keys, and relationships

Logical Schema (Core Entities)

Table 3.1 summarizes entities, salient attributes, and relationships.

Table 3.1: Entity summary (compact)

Entity	Key Attributes	Relationships
User	user_id (PK, UUID), email, timezone	1–N ContentItem, 1–N PlatformIntegration
PlatformIntegration	integration_id (PK), platform (ENUM), accessTokenEnc	N–1 User
ContentItem	content_id (PK), type (ENUM), status (ENUM), title	N–1 User; 1–N DraftVersion; 1–N ScheduleJob; N–M ImageAsset (via ContentImageLink)
DraftVersion	version_id (PK), bodyHtml, createdAt	N–1 ContentItem
ImageAsset	image_id (PK), storageUrl, altText	N–M ContentItem (via ContentImageLink)
ContentImageLink	link_id (PK), role (ENUM), ordering	N–1 ContentItem, N–1 ImageAsset
ScheduleJob	job_id (PK), platform (ENUM), scheduled_time, status	N–1 ContentItem; 0..1–1 PublishResult
PublishResult	result_id (PK), platformPostId, permalink, status	1–1 ScheduleJob

Bibliography