Functional Specification: PromptIt Version 0.2 (Cadence)

1. Executive Summary

Project: PromptIt (*Prompt It!*)
Description: Web Browser Extension targeting Chromium-based browsers (Google Chrome, Microsoft Edge, Brave, Opera) enabling AI text transformation of web content.
Version: 0.2
Codename: Cadence

PromptIt Version 0.2, codenamed **Cadence**, is a major stability and control release. The primary focus of this update is to introduce **per-promptlet token management** for enhanced cost control and feature reliability, while stabilizing the core CRUD and execution logic. "Cadence" reflects the new, measured, and controlled execution flow, particularly concerning the cost of API calls. The release resolves critical bugs related to the default promptlet reset and editor initialization, ensuring a seamless user experience.

2. Features and Scope

2.1. Core Functionality (Retained)

Feature	Description
Context Menu Execution	Users can select any text on a webpage and run a promptlet via the right-click context menu.

Side Panel Output	All AI responses are displayed in a dedicated Chrome Side Panel for non-intrusive viewing and interaction.
Promptlet CRUD	Users can Create, Read, Update, and Delete custom promptlets via the **Manage Promptlets** options page.
API Key Management	Secure storage and utilization of the user's OpenAI API Key.
Chaining/Re-prompting	Users can immediately re-run a promptlet on the generated output or a new selection from the side panel.

2.2. Version 2.0 Cadence: New and Enhanced Features

Feature	Scope
Per-Promptlet Max Tokens (NEW)	The promptlet data model now includes a maxTokens field. This allows default promptlets like "Verify" to use a higher limit (e.g., 4000) for complex tasks, while custom prompts default to a cost-conscious limit (1500).
Default Model Update	The system default model is updated to **GPT-4o**, reflecting the new standard for high-quality, efficient AI interaction.
Max Tokens Editor Range	The Max Tokens editor in manage.html is updated with a maximum value of 16,000 to support the full range of modern models.

Reset/Editor Stability (FIXED)	Critical bugs related to resetting default promptlets and initializing the editor for new/cloned promptlets have been resolved, ensuring the management page is robust.

3. Design Philosophy

3.1. Principle: Contextual Utility

The primary design goal is to provide immediate, powerful AI utility exactly where the user needs it: **on the selected text**. By utilizing the context menu and the side panel, the extension minimizes workflow interruption.

3.2. Principle: Controlled Power

The Cadence release heavily emphasizes **Controlled Power**. By introducing configurable maxTokens, the extension balances the need for powerful, long-form AI responses (e.g., for detailed analysis or verification) with the need for cost awareness, which is addressed by defaulting new custom promptlets to a lower token count.

3.3. Principle: Extensibility

Promptlets are designed as a flexible data structure that allows for easy future expansion of advanced settings (e.g., Top P, Frequency Penalty) and structured output definitions (outputStructure array).

4. Technical Architecture and Data Model

4.1. Architecture Components

1. **Background Service Worker (background.js):**
   - **Lifetime:** Event-driven, typically idle until an event (e.g., right-click, message).
   - **Responsibilities:** Initialization, API Key check, Context Menu creation/rebuilding, and routing messages between the side panel and the management page.
2. **Side Panel (sidepanel.html, sidepanel.js):**
   - **Responsibilities:** Receives the selected text and promptlet definition, executes the asynchronous API call, and displays the final output. Handles error reporting (e.g., missing API Key).
3. **Management Page (manage.html, manage.js):**
   - **Responsibilities:** Provides the user interface for **C**reate, **R**ead, **U**pdate, **D**elete (CRUD) operations for promptlets. Also handles API Key input.

4.2. Promptlet Data Model

Each promptlet object is stored locally in Chrome's storage and includes the following fields:

Field Name	Type	Description	Default Value (New)
name	String	User-facing name of the promptlet.	N/A
emoji	String	Icon for display in the context menu and editor.	📝
prompt	String	The core instruction sent to the AI model.	N/A
model	String	The specific AI	gpt-4o

		model to use (e.g., gpt-4o).	
temperature	Number	Controls randomness (creativity). Range: 0.0 to 2.0.	1.0
maxTokens	**Number**	**Maximum length of the AI response.** (NEW in v2.0)	**1500**
outputStructure	Array	Defines expected output sections (e.g., ["main", "notes"]).	["main"]
isDefault	Boolean	System flag; prevents deletion of default promptlets.	false

5. User Guidelines

5.1. Initial Setup

1. **API Key:** Navigate to the **Manage Promptlets** options page and enter a valid OpenAI API Key in the dedicated section. Without an API Key, the extension cannot function.
2. **Default Promptlets:** On first install, the extension populates a set of optimized default promptlets (e.g., "Summarise," "Verify") that are immediately available in the context menu.

5.2. Running a Promptlet

1. **Selection:** Select any text on a webpage.
2. **Access:** Right-click the selected text to open the context menu.
3. **Execute:** Hover over the **Prompt It!** root menu and select the desired promptlet (e.g., "Rephrase").
4. **View Output:** The Side Panel will automatically open, display a processing status, and then present the AI-generated result.

5.3. Managing Promptlets

- **Access:** Select ⚙️ **Manage Promptlets** from the context menu.
- **Editing/Cloning:** Click the **Edit** or **Clone** button on any promptlet card to open the editor.
  - Use the **Advanced Settings** toggle to modify the model, temperature, and the new **Max Tokens** setting.
- **Cost Control:** When creating a new promptlet, the **Max Tokens** value defaults to 1500. Adjust this value only if you require longer responses (e.g., detailed reports), as higher values increase API call costs.
- **Reset:** The **Reset Default Promptlets** button will restore the original, optimized default set, overwriting any custom changes to those defaults.