Technical Documentation – Book Creator App & Make.com Scenario

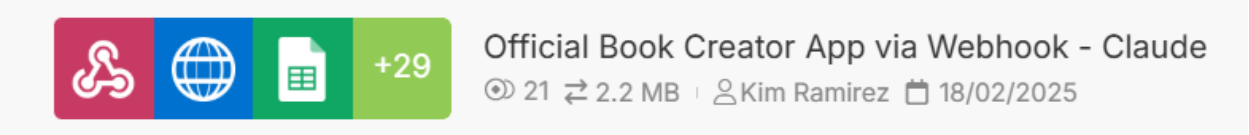
1. Overview

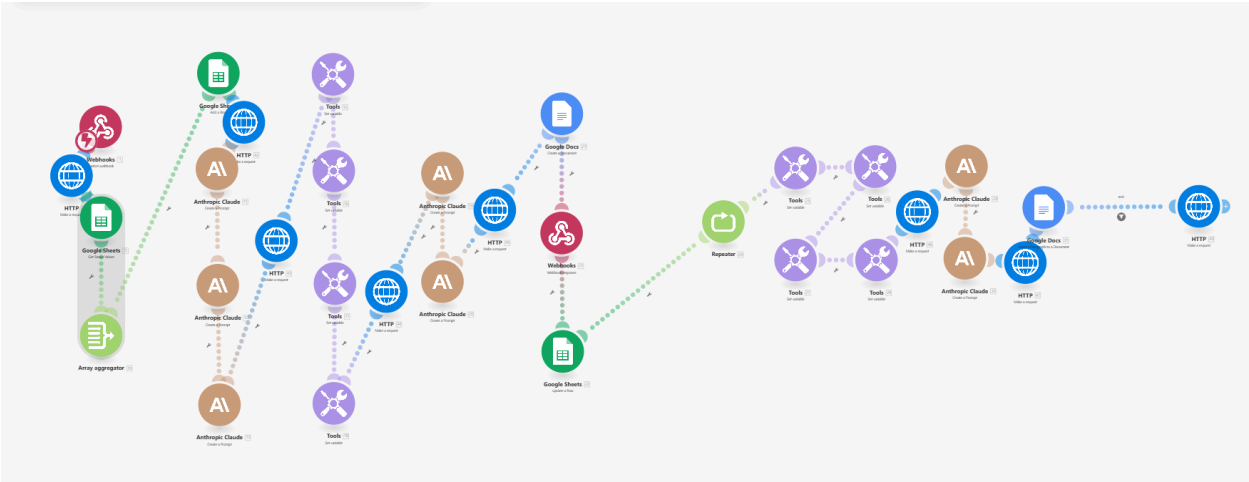
The **Book Creator App** automates book generation using AI (Anthropic Claude, Gemini, and Perplexity AI) and integrates with **Make.com** to handle workflow automation between the frontend, Google Sheets, Google Docs, and webhooks.

This documentation outlines how the system works and what the next developer needs to know to update both the **frontend code** and the **Make.com scenario**.

How to access the scenario:

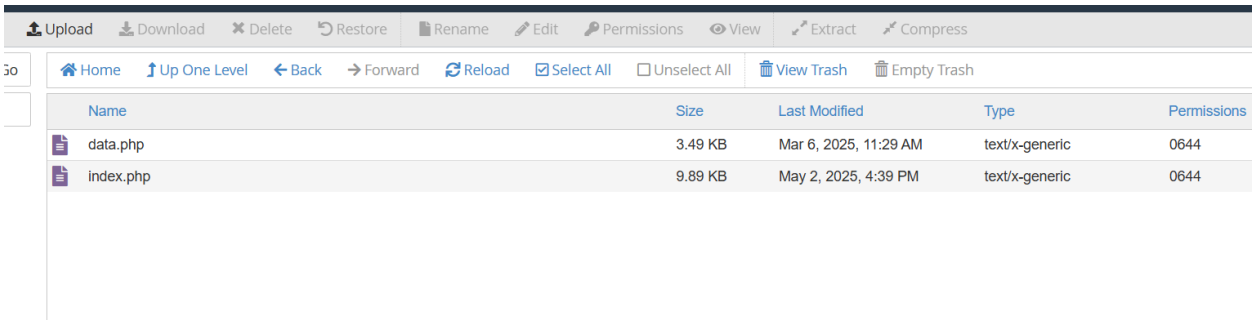
1. Log in to make.com (make sure that your account is associated as a member to Unica's organization on the platform).
2. Select Official Book Creator App via Webhook - Claude as the default scenario. There's another scenario, which is for Gemini and Perplexity models.





How to access the front-end source code:

1. Login to unicacloud.me/cpanel (Credentials: unicacloud/RRScRhlrxrb2)
2. Go to public_html, then navigate to the directory called **make**.



---

2. System Architecture

Main Components

Component	Purpose
Frontend (Book Creator App)	HTML-based form for user input and sending data to Make.com webhook
Make.com Scenario	Automation workflow connecting AI, Google Sheets, Google Docs, and HTTP requests
Anthropic Claude API	Used to generate book chapters, summaries, or full manuscripts
Google Sheets	Used for logging requests, tracking metadata, and storing generation parameters
Google Docs	Stores the final AI-generated book output
Webhook (Make.com)	Receives data from the frontend and triggers the automation chain

---

3. Frontend (Book Creator App)

3.1. UI Components

- **Fields:**
    - `Book Title`
    - `Book Topic`
    - `What Book is About`
    - `Number of Chapters`
    - `Additional Detail`
    - `Choose AI` (Default: Claude)
    - `Output`: Google Docs link (auto-filled after generation)
- **Buttons:**
    - `Send`: Sends form data to Make.com webhook.
    - `Create New Book`: Clears all inputs for a new session.

3.2. API Integration

- **Webhook URL:**
  Defined inside the frontend script (e.g., `https://hook.make.com/<unique_id>`).
- **HTTP Method:**
  `POST`

Payload Example:

```
{
  "book_title": "The Future of AI",
  "book_topic": "Artificial Intelligence",
  "book_about": "Explores AI in education",
  "number_of_chapters": 10,
  "additional_detail": "Add case studies and interviews",
  "chosen_ai": "Claude"
}
```

-

Expected Response:

```
{
  "google_docs_link": "<Google Docs link of the book>"
}
```

-

3.3. Update Instructions

- Locate the `send` button event handler.
- Update the **Make.com webhook URL** if a new one is created.
- Ensure **CORS settings** are correctly configured for Make.com endpoints.
- Validate form data before sending to prevent empty submissions.

---

4. Make.com Scenario Workflow

4.1. General Flow

1. **Webhook Trigger** – Receives payload from frontend.
2. **Google Sheets (Log Entry)** – Records input data.
3. **Array Aggregator** – Consolidates input fields.
4. **Anthropic Claude Modules** – Generate:
    - Chapter outlines
    - Chapter content
    - Summaries
5. **HTTP Modules** – Send/receive API data to and from Claude.
6. **Google Docs** – Create or update a new document with the generated text.
7. **Google Sheets (Final Update)** – Add Google Docs link.
8. **Repeater** – Iterates over chapters if multiple are requested.
9. **Webhook Response** – Sends Google Docs link back to the frontend.

4.2. Key Modules in the Screenshot

Module	Description
Webhook (Trigger)	Entry point from frontend
Google Sheets	Logs request metadata
Array Aggregator	Collects user input values for AI generation
Anthropic Claude (x5)	Multiple steps for generating book sections
Tools (Iterator / Text)	Splits, merges, and formats text between steps
Google Docs	Creates the final book output
Repeater	Loops through each chapter generation process
HTTP (blue icons)	Handles Claude API requests or Make.com internal calls

Final Webhook Response	Returns generated Google Docs URL

4.3. Data Mapping Summary

Source	Destination	Purpose
Webhook Input	Google Sheets	Logging book metadata
Sheets Data	Claude	Provides context for book generation
Claude Output	Google Docs	Creates content document
Docs URL	Webhook Response	Returned to frontend
Repeater	Claude	Iterative generation for multi-chapter books

4.4. Claude Prompt Template Example

Each Claude module typically uses a structured prompt like:

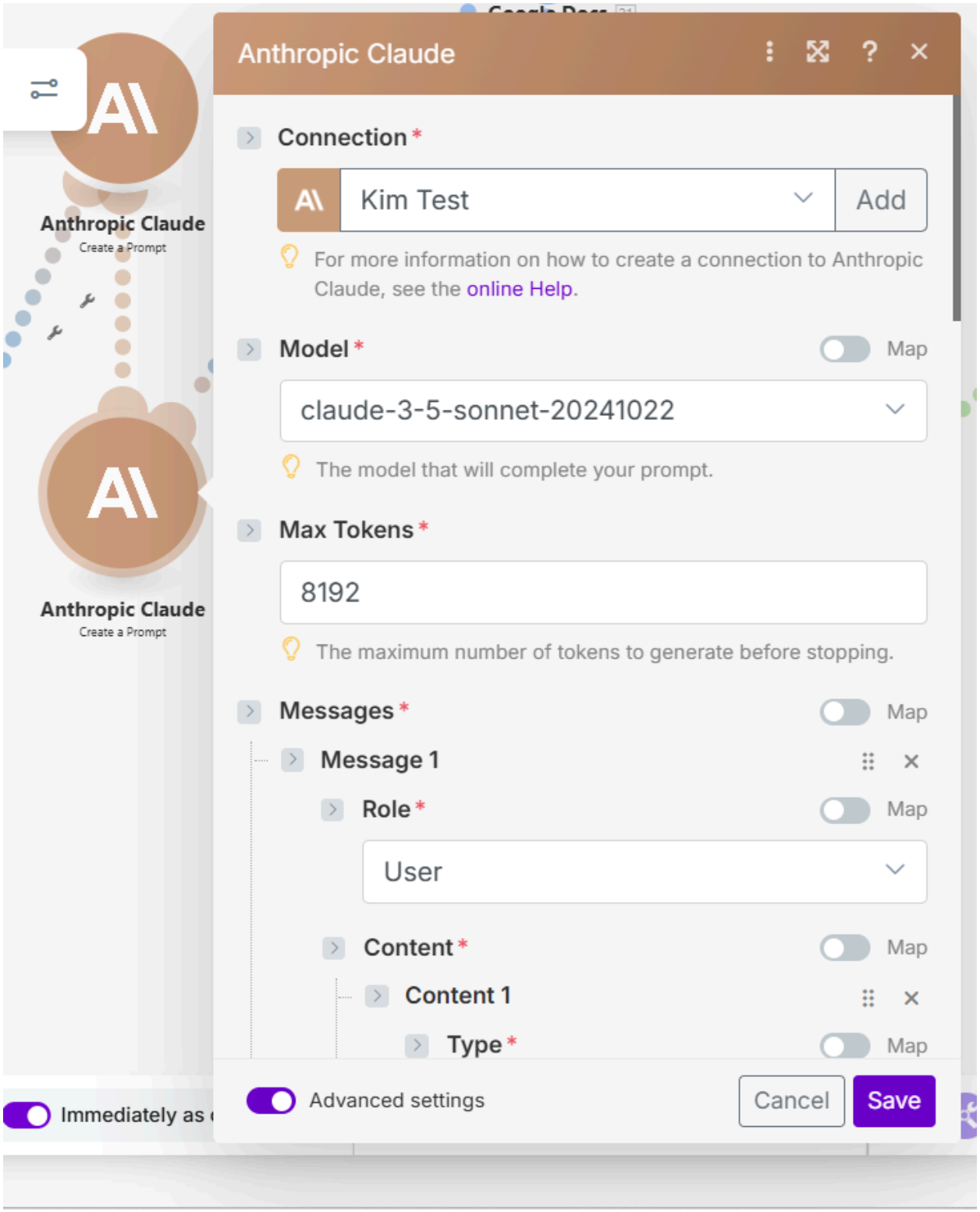
```
You are a professional book ghostwriter.
Title: {{Book Title}}
Topic: {{Book Topic}}
Description: {{Book About}}
Chapters: {{Number of Chapters}}
Additional Detail: {{Additional Detail}}

Generate a detailed outline or content for Chapter {{n}}.
```

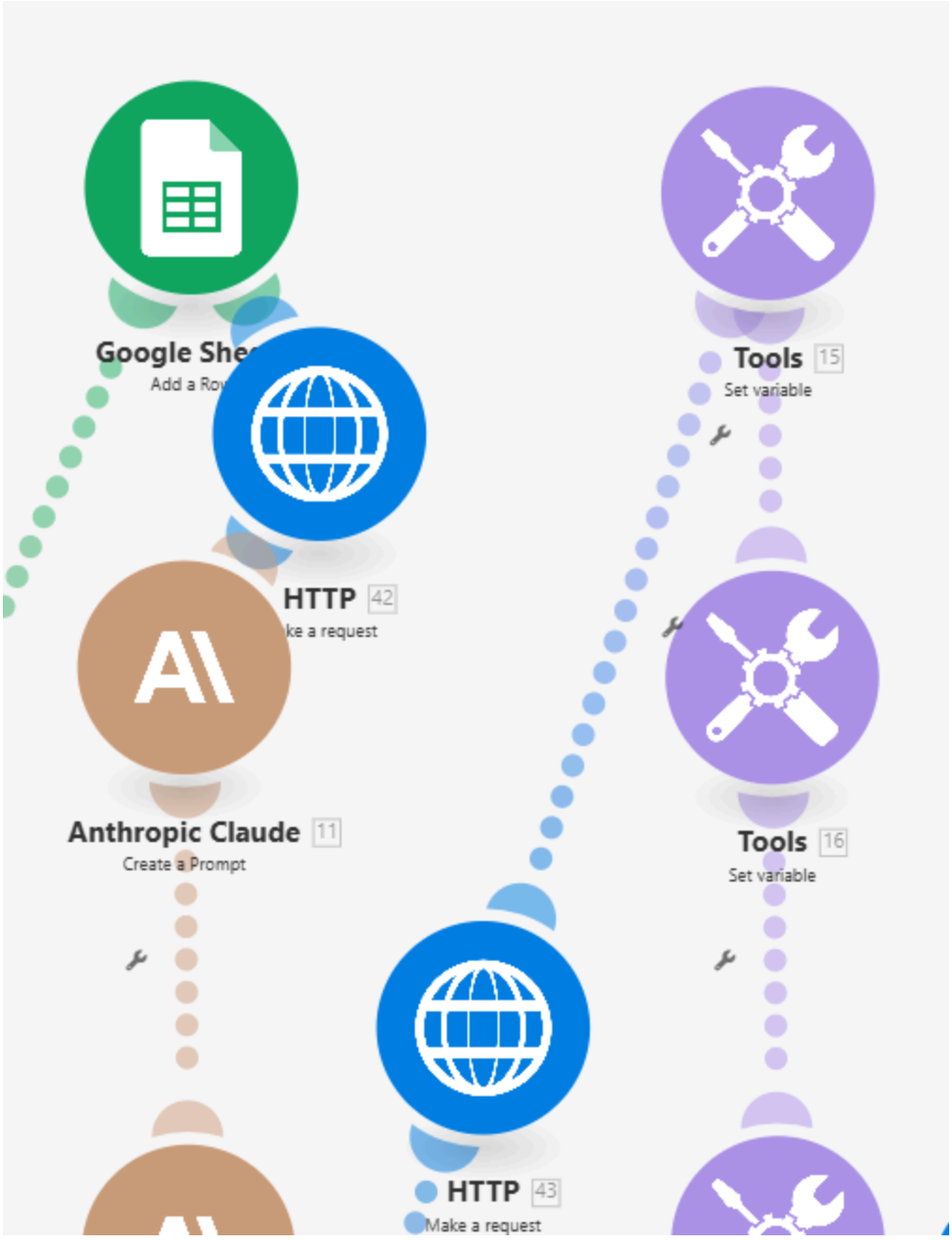
> Note: Prompts can be found and edited inside each **Anthropic Claude** module in Make.com.

4.4. Nodes/Modules Components and Functions

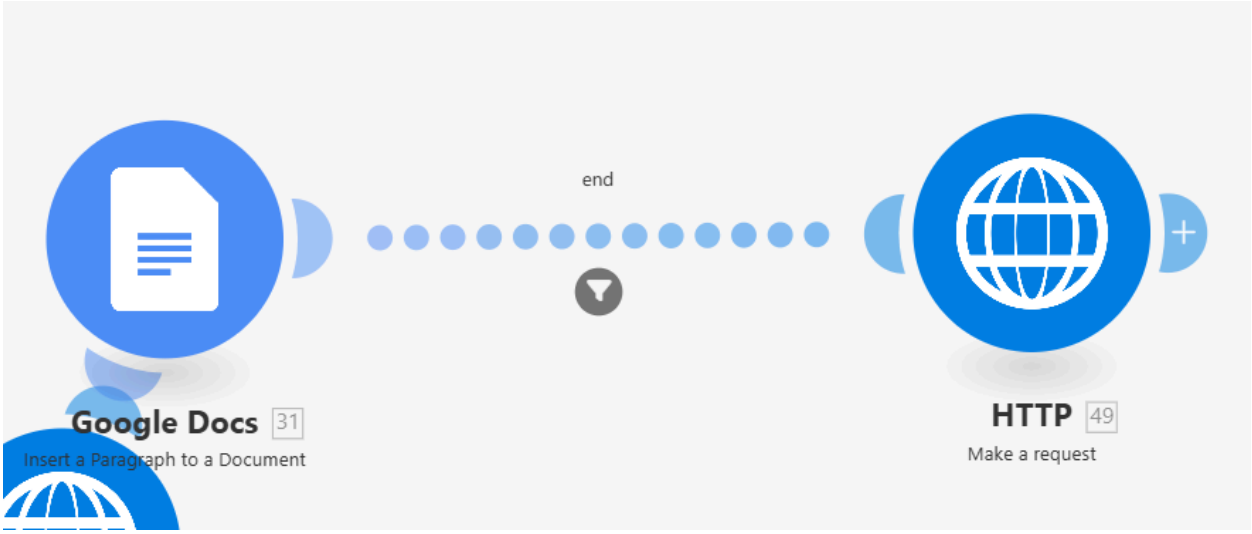
1. Each Make.com module/node has its settings interface so that you can set up API connections, etc. Just click the node will setup.

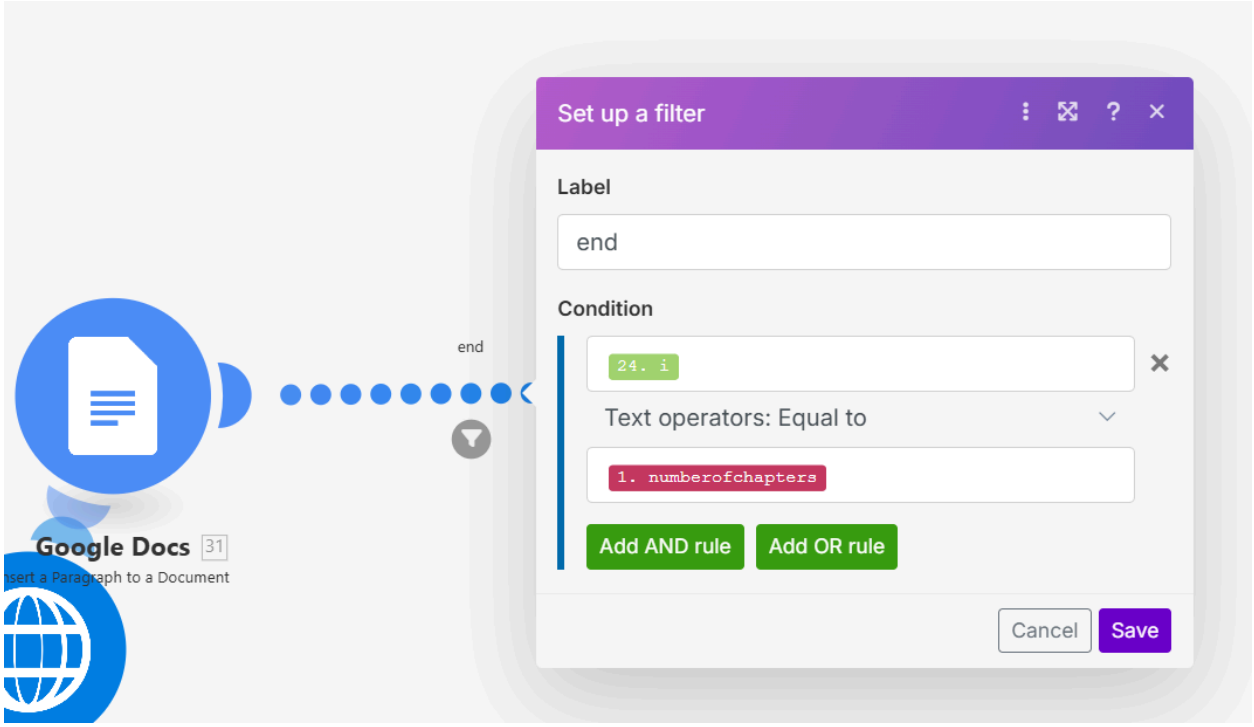


2. Every node also has connectors so that the workflow sequence will be determined.



3. Filters: You can set up a filter based on the logic you need to apply.





---

5. Maintenance Guidelines

To Update AI Logic

- Go to the **Anthropic Claude** modules in Make.com.
- Adjust prompts for tone, format, or structure.
- If switching to another AI, update the **Choose AI** logic in frontend and Make.com router.

To Update Google Sheets / Docs Connections

- Re-authenticate Google account if credentials expire.
- Verify the target Google Sheet and Doc templates exist.
- Make sure the column headers in Google Sheets match the JSON payload.

To Update the Webhook

- Create a new Webhook in Make.com (if needed).
- Replace the webhook URL in the frontend code.
- Test the connection by sending a sample request.

To Debug Scenario

1. Enable “**Log Execution**” in Make.com.
2. Test using sample data from frontend.
3. Check for failed modules (red icons) and review the error logs.
4. Re-run the scenario manually if partial failures occur.

---

6. Version Control & Deployment

Component	Recommended Tool	Notes
Frontend	GitHub / GitLab	Commit after every UI or logic change
Make.com Scenario	Duplicate before editing	Keep versioned copies for rollback
Google Sheets & Docs	Shared Drive	Maintain access permissions for service accounts

---

7. Future Improvements

- Add **progress indicators** to frontend while book generation runs.
- Include **user authentication** for tracking generated books.
- Support **multiple AI models** dynamically (Claude, GPT, Gemini, etc.).
- Store all generated data in a backend database for analytics.

Document Author: Kim Ramirez
Last Updated: October 2025
Version: 1.0