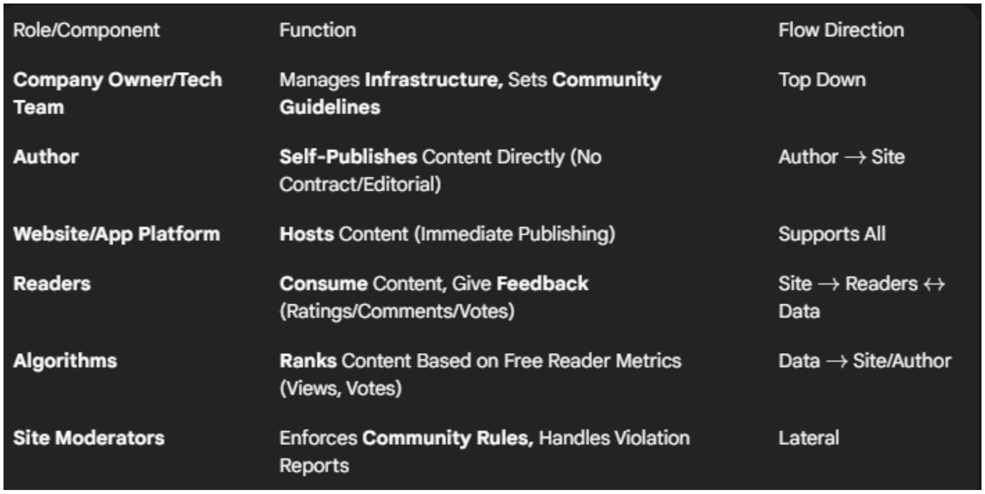
CIS 3100 - Management Information Systems

Organizational Chart

- Identify the key flow of managerial instruction, employee output, and reporting.

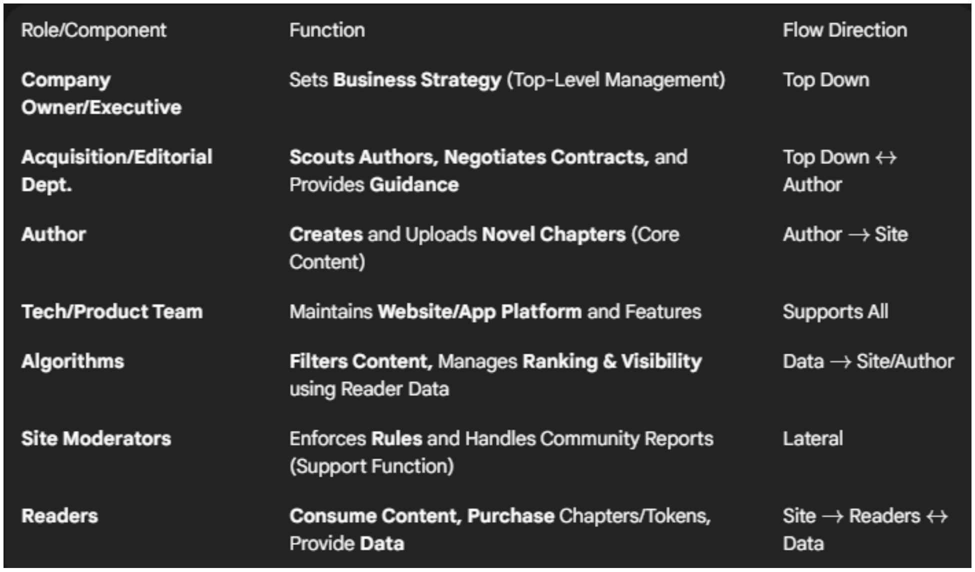
There are two main types of novel sites, ignoring novel cataloging websites, such as Goodreads or NovelUpdates.

Free Site:
Site Owner → Site Moderator → Upload Website→ Authors → Readers

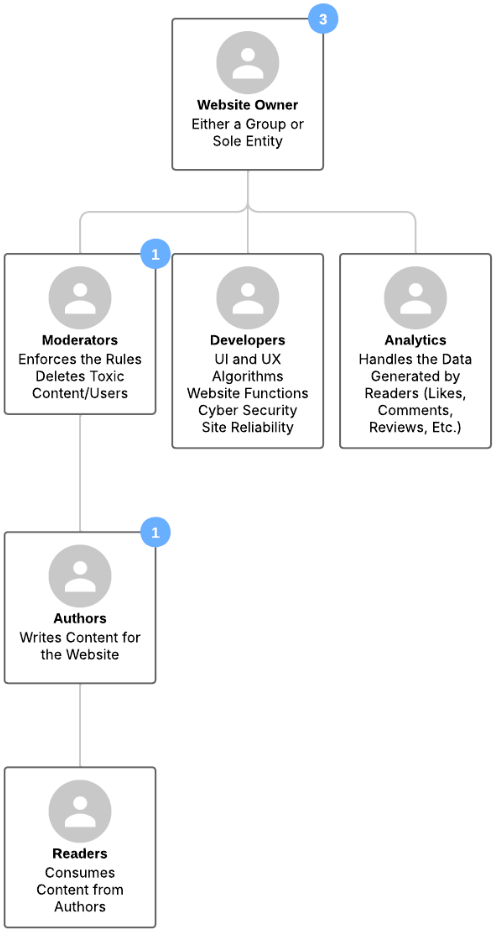
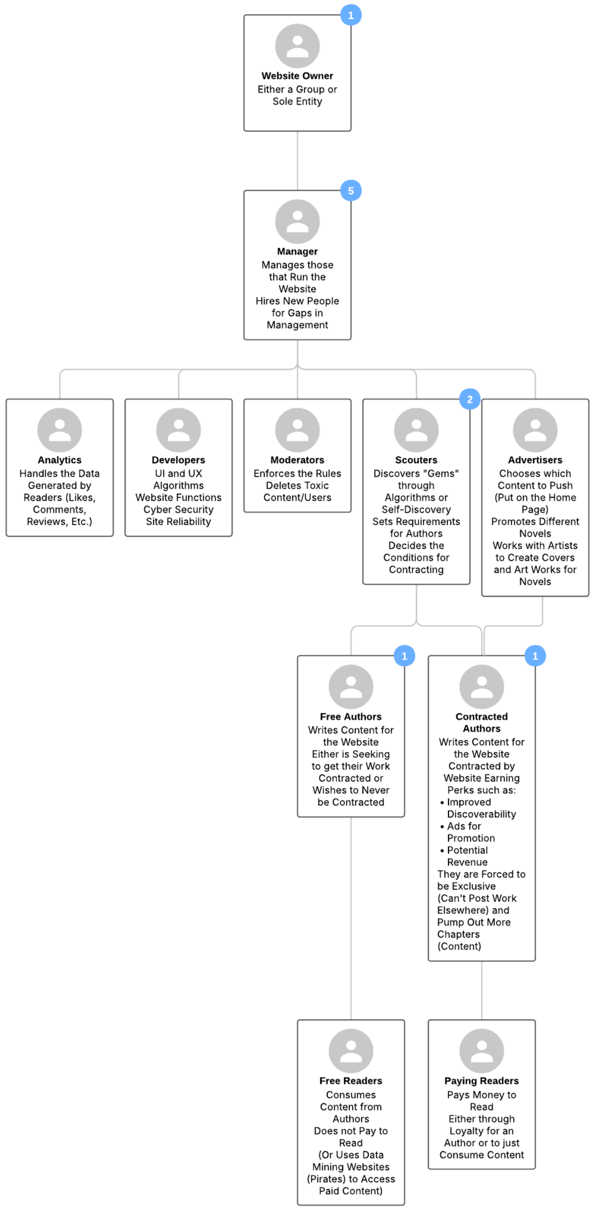


Role/Component	Function	Flow Direction
Company Owner/Tech Team	Manages **Infrastructure,** Sets **Community Guidelines**	Top Down
Author	**Self-Publishes** Content Directly (No Contract/Editorial)	Author → Site
Website/App Platform	**Hosts** Content (Immediate Publishing)	Supports All
Readers	**Consume** Content, Give **Feedback** (Ratings/Comments/Votes)	Site → Readers ↔ Data
Algorithms	**Ranks** Content Based on Free Reader Metrics (Views, Votes)	Data → Site/Author
Site Moderators	Enforces **Community Rules,** Handles Violation Reports	Lateral

Contracting Site:
Company Owner (Novel Publishing [Online]) → Site Moderators → Website → Contractors and Talent Scouts (Mostly Algorithm) → Author → Novels → Readers

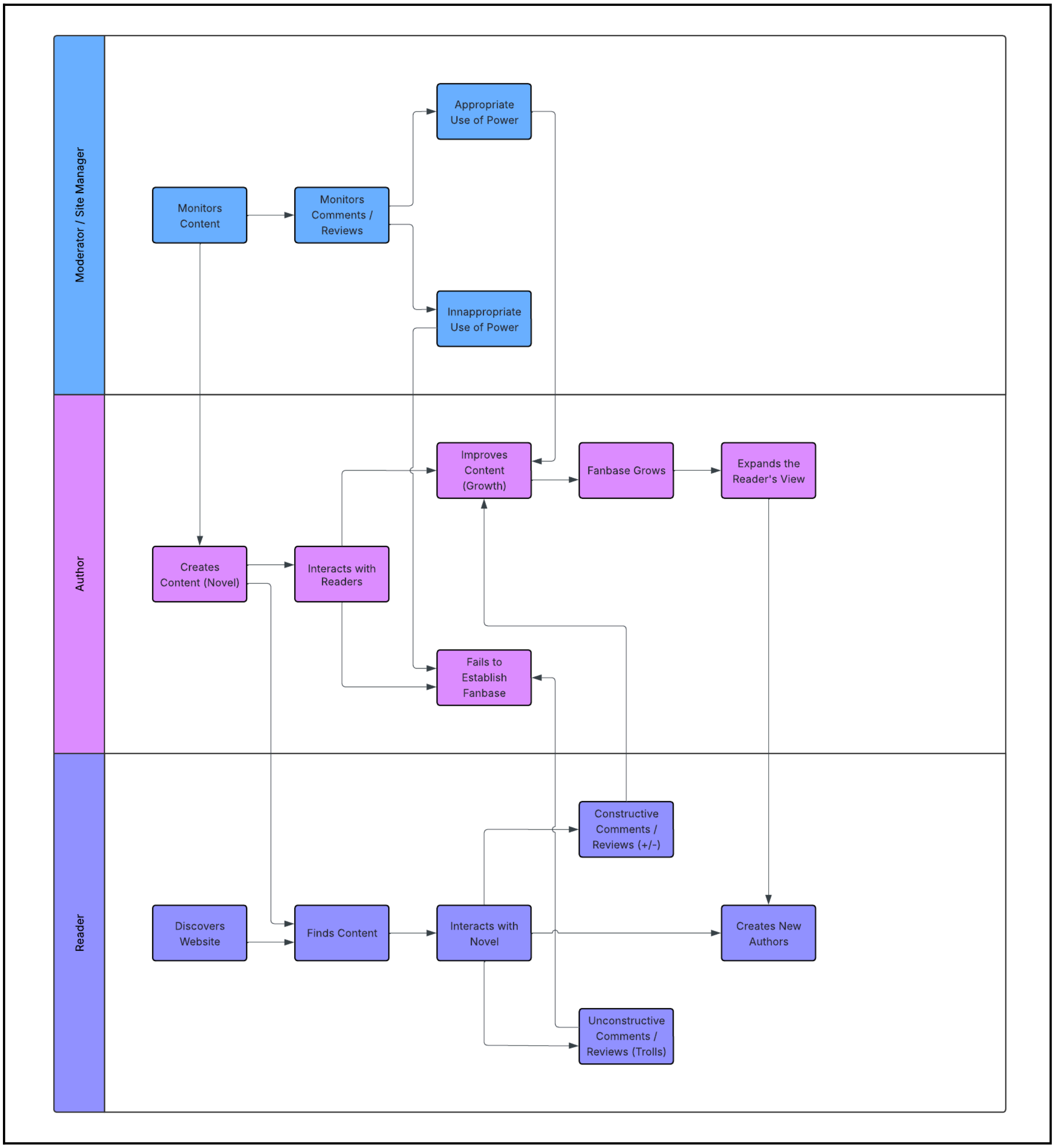


Role/Component	Function	Flow Direction
Company Owner/Executive	Sets **Business Strategy** (Top-Level Management)	Top Down
Acquisition/Editorial Dept.	**Scouts Authors, Negotiates Contracts,** and Provides **Guidance**	Top Down ↔ Author
Author	**Creates** and Uploads **Novel Chapters** (Core Content)	Author → Site
Tech/Product Team	Maintains **Website/App Platform** and Features	Supports All
Algorithms	**Filters Content,** Manages **Ranking & Visibility** using Reader Data	Data → Site/Author
Site Moderators	Enforces **Rules** and Handles Community Reports (Support Function)	Lateral
Readers	**Consume Content, Purchase** Chapters/Tokens, Provide **Data**	Site → Readers ↔ Data

Free Sites	Contracting Sites
 3 **Website Owner** Either a Group or Sole Entity 1 **Moderators** Enforces the Rules Deletes Toxic Content/Users **Developers** UI and UX Algorithms Website Functions Cyber Security Site Reliability **Analytics** Handles the Data Generated by Readers (Likes, Comments, Reviews, Etc.) 1 **Authors** Writes Content for the Website **Readers** Consumes Content from Authors	 1 **Website Owner** Either a Group or Sole Entity 5 **Manager** Manages those that Run the Website Hires New People for Gaps in Management **Analytics** Handles the Data Generated by Readers (Likes, Comments, Reviews, Etc.) **Developers** UI and UX Algorithms Website Functions Cyber Security Site Reliability **Moderators** Enforces the Rules Deletes Toxic Content/Users 2 **Scouters** Discovers "Gems" through Algorithms or Self-Discovery Sets Requirements for Authors Decides the Conditions for Contracting **Advertisers** Chooses which Content to Push (Put on the Home Page) Promotes Different Novels Works with Artists to Create Covers and Art Works for Novels 1 **Free Authors** Writes Content for the Website Either is Seeking to get their Work Contracted or Wishes to Never be Contracted 1 **Contracted Authors** Writes Content for the Website Contracted by Website Earning Perks such as: • Improved Discoverability • Ads for Promotion • Potential Revenue They are Forced to be Exclusive (Can't Post Work Elsewhere) and Pump Out More Chapters (Content) **Free Readers** Consumes Content from Authors Does not Pay to Read (Or Uses Data Mining Websites (Pirates) to Access Paid Content) **Paying Readers** Pays Money to Read Either through Loyalty for an Author or to just Consume Content

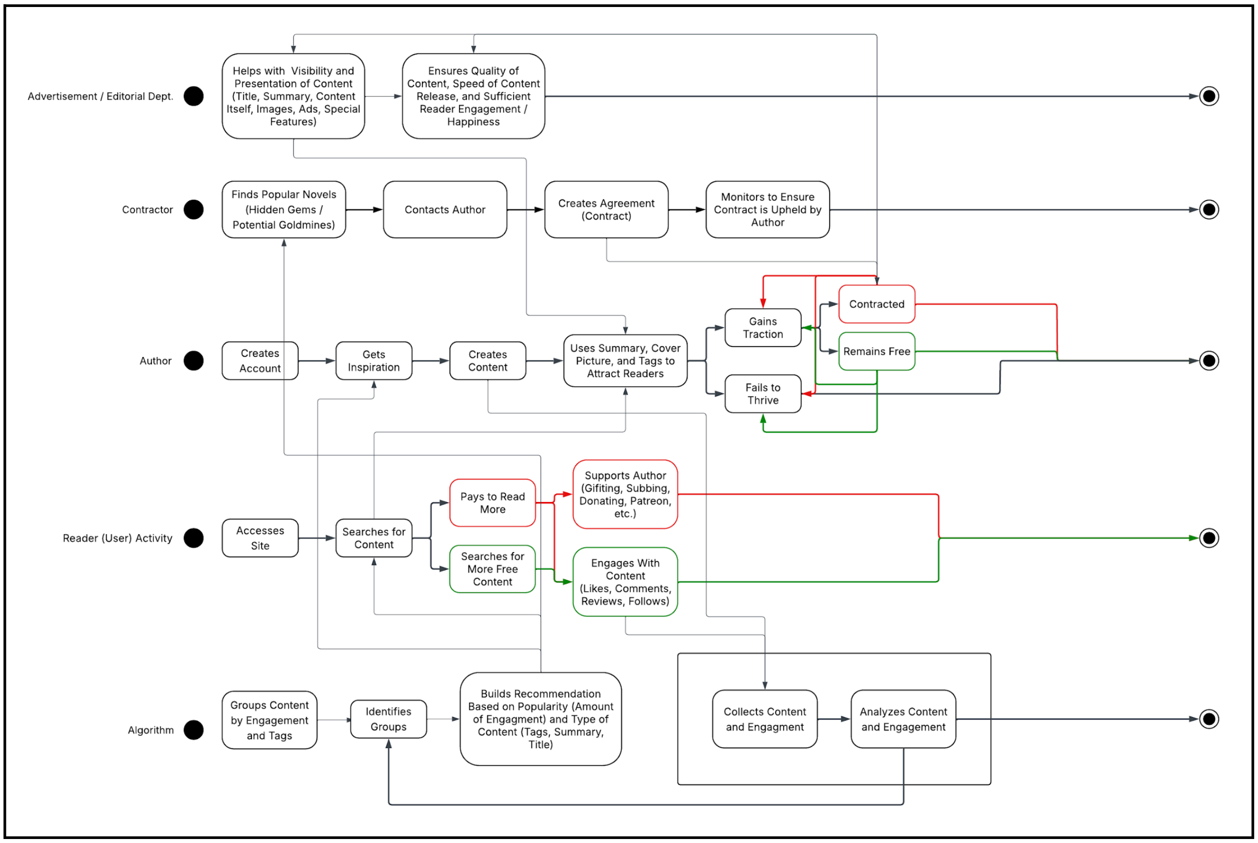
Business Process Model Notation

- Compose a final graphic using a diagram tool.


Moderator / Site Manager
Monitors Content
Monitors Comments / Reviews
Appropriate Use of Power
Innappropriate Use of Power
Author
Creates Content (Novel)
Interacts with Readers
Improves Content (Growth)
Fanbase Grows
Expands the Reader's View
Fails to Establish Fanbase
Reader
Discovers Website
Finds Content
Interacts with Novel
Constructive Comments / Reviews (+/-)
Unconstructive Comments / Reviews (Trolls)
Creates New Authors

Activity Diagram

- Using a diagram tool.

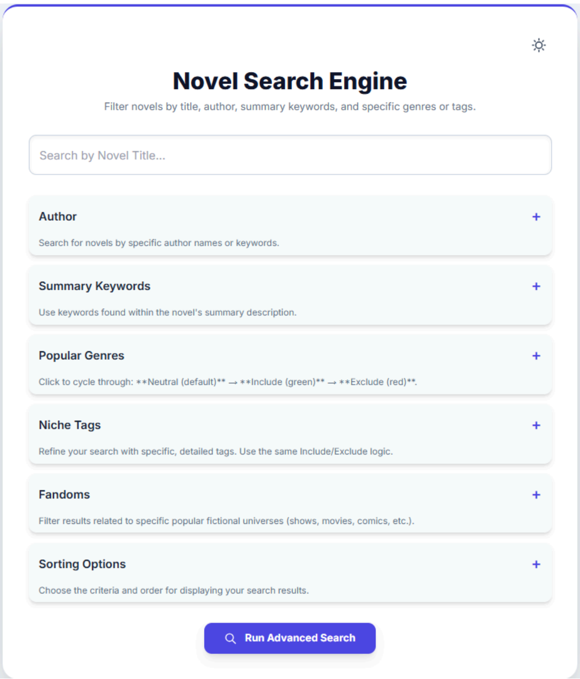

Advertisement / Editorial Dept.
Helps with Visibility and Presentation of Content (Title, Summary, Content Itself, Images, Ads, Special Features)
Ensures Quality of Content, Speed of Content Release, and Sufficient Reader Engagement / Happiness
Contractor
Finds Popular Novels (Hidden Gems / Potential Goldmines)
Contacts Author
Creates Agreement (Contract)
Monitors to Ensure Contract is Upheld by Author
Author
Creates Account
Gets Inspiration
Creates Content
Uses Summary, Cover Picture, and Tags to Attract Readers
Gains Traction
Fails to Thrive
Contracted
Remains Free
Reader (User) Activity
Accesses Site
Searches for Content
Pays to Read More
Searches for More Free Content
Supports Author (Gifiting, Subbing, Donating, Patreon, etc.)
Engages With Content (Likes, Comments, Reviews, Follows)
Algorithm
Groups Content by Engagement and Tags
Identifies Groups
Builds Recommendation Based on Popularity (Amount of Engagment) and Type of Content (Tags, Summary, Title)
Collects Content and Engagment
Analyzes Content and Engagement

UI/UX Wire Frame

- Use a diagram tool (UI composer).

GitHub Link: [Advanced Novel Search Interface]()

Screenshots:



Novel Search Engine

Filter novels by title, author, summary keywords, and specific genres or tags.

Search by Novel Title...

Author +

Search for novels by specific author names or keywords.

Summary Keywords +

Use keywords found within the novel's summary description.

Popular Genres +

Click to cycle through: **Neutral (default)** → **Include (green)** → **Exclude (red)**.

Niche Tags +

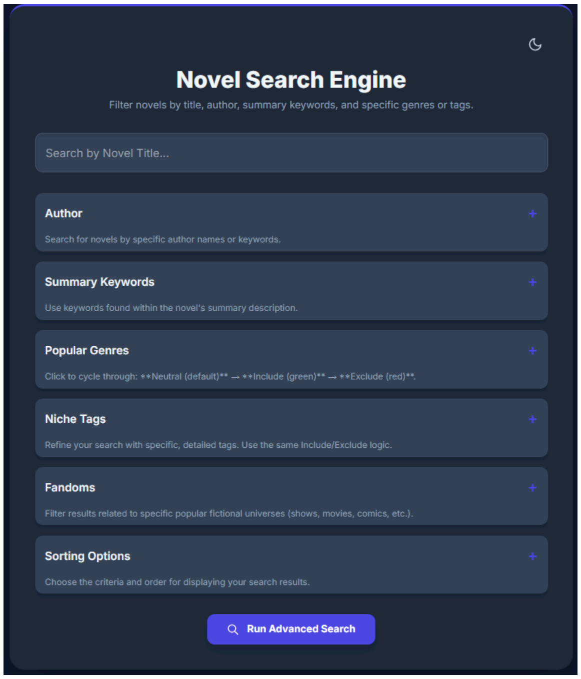
Refine your search with specific, detailed tags. Use the same Include/Exclude logic.

Fandoms +

Filter results related to specific popular fictional universes (shows, movies, comics, etc.).

Sorting Options +

Choose the criteria and order for displaying your search results.



Novel Search Engine

Filter novels by title, author, summary keywords, and specific genres or tags.

Search by Novel Title...

Author +

Search for novels by specific author names or keywords.

Summary Keywords +

Use keywords found within the novel's summary description.

Popular Genres +

Click to cycle through: **Neutral (default)** → **Include (green)** → **Exclude (red)**.

Niche Tags +

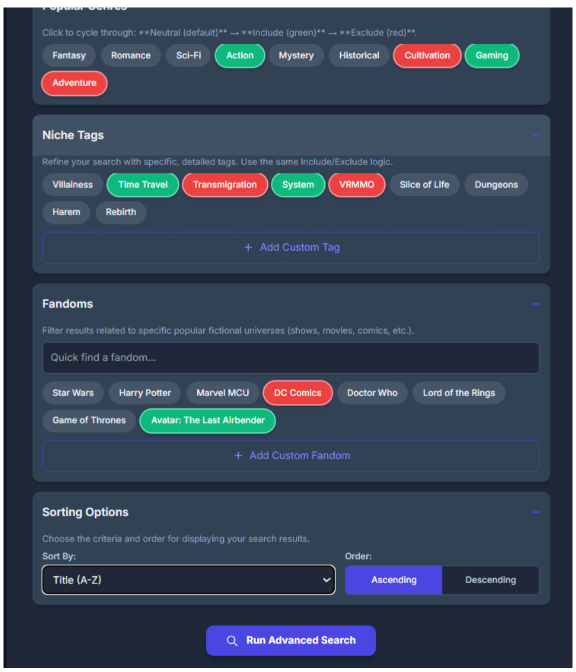
Refine your search with specific, detailed tags. Use the same Include/Exclude logic.

Fandoms +

Filter results related to specific popular fictional universes (shows, movies, comics, etc.).

Sorting Options +

Choose the criteria and order for displaying your search results.



Click to cycle through: **Neutral (default)** → **Include (green)** → **Exclude (red)**.

Niche Tags –

Refine your search with specific, detailed tags. Use the same Include/Exclude logic.

Fandoms –

Filter results related to specific popular fictional universes (shows, movies, comics, etc.).

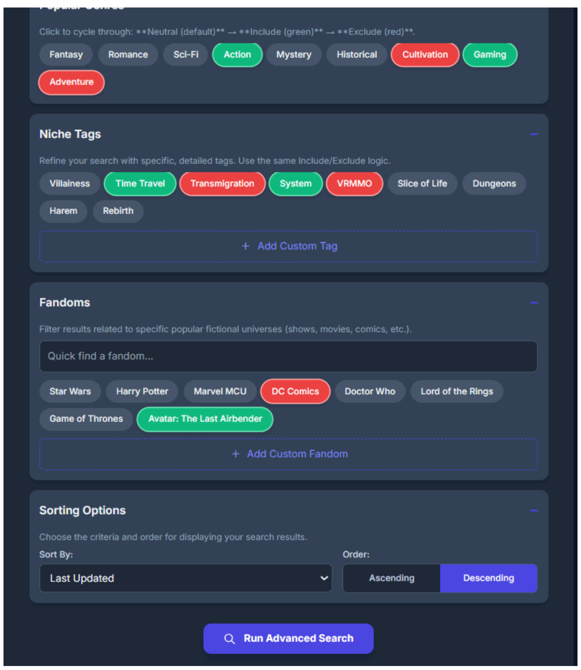
Quick find a fandom...

Sorting Options –

Choose the criteria and order for displaying your search results.

Sort By: Title (A-Z)

Order: Ascending | Descending



Click to cycle through: **Neutral (default)** → **Include (green)** → **Exclude (red)**.

Niche Tags –

Refine your search with specific, detailed tags. Use the same Include/Exclude logic.

Fandoms –

Filter results related to specific popular fictional universes (shows, movies, comics, etc.).

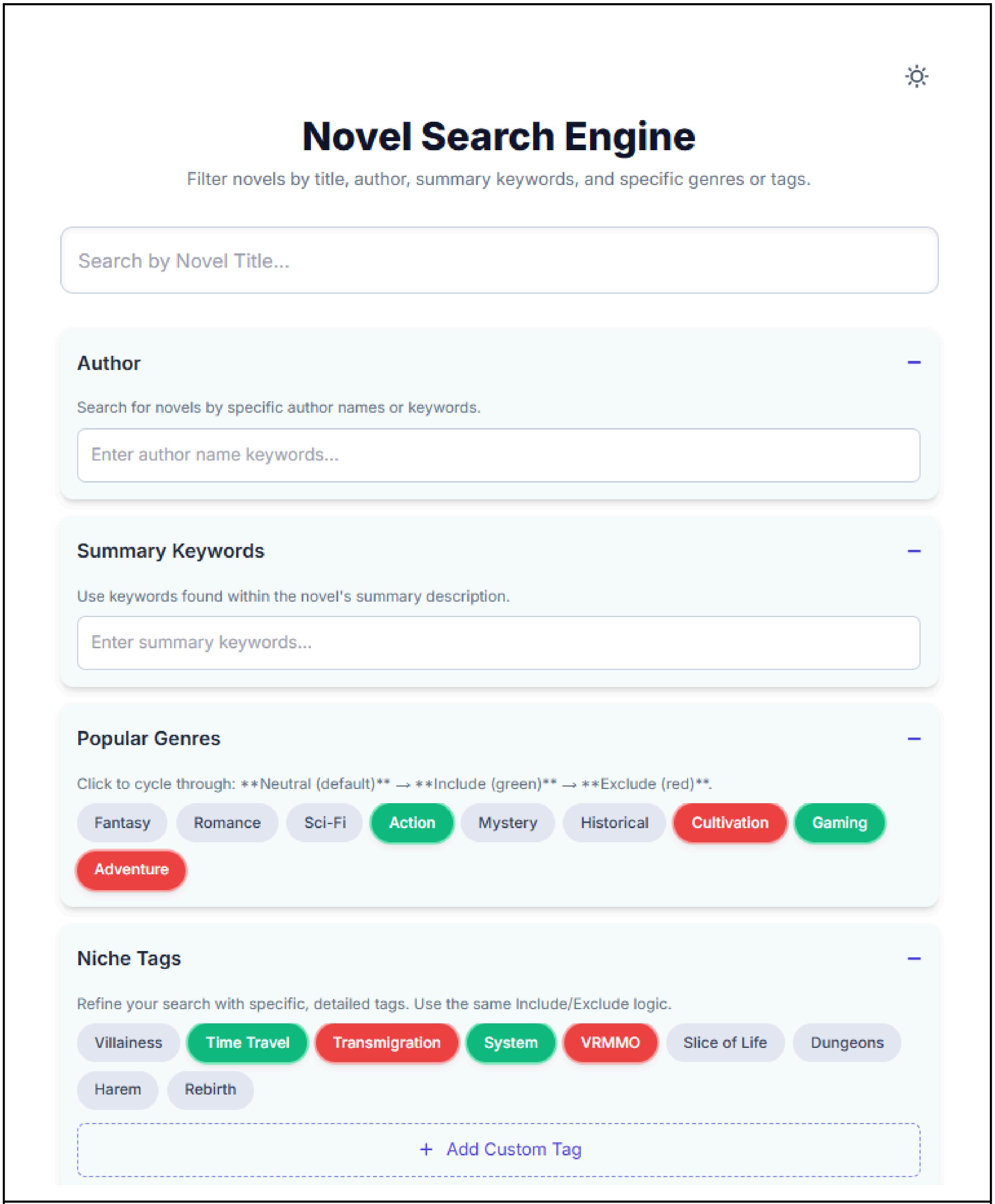
Quick find a fandom...

Sorting Options –

Choose the criteria and order for displaying your search results.

Sort By: Last Updated

Order: Ascending | Descending



Novel Search Engine

Filter novels by title, author, summary keywords, and specific genres or tags.

Search by Novel Title...

Author

Search for novels by specific author names or keywords.

Enter author name keywords...

Summary Keywords

Use keywords found within the novel's summary description.

Enter summary keywords...

Popular Genres

Click to cycle through: **Neutral (default)** → **Include (green)** → **Exclude (red)**.

Fantasy Romance Sci-Fi Action Mystery Historical Cultivation Gaming Adventure

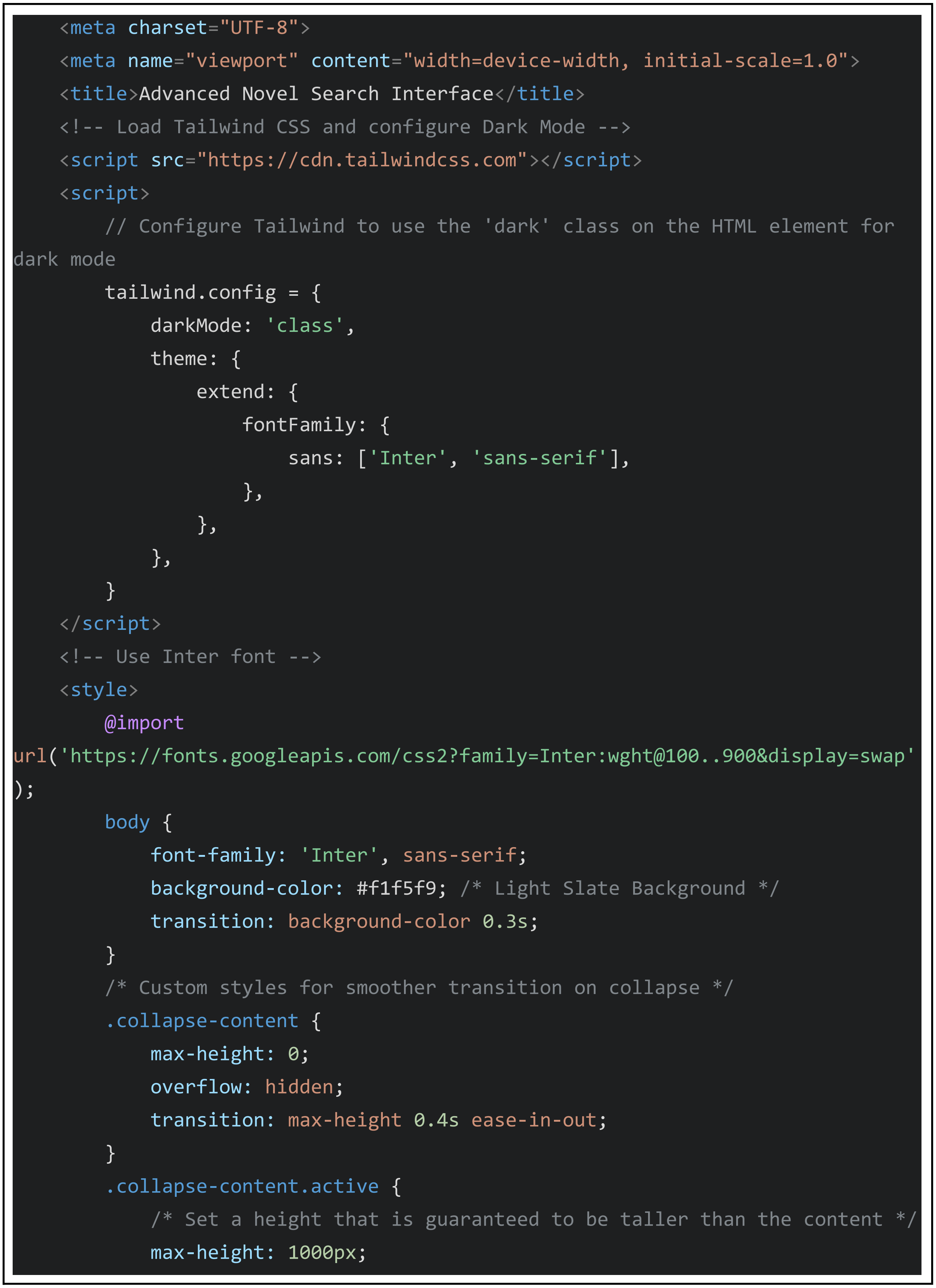
Niche Tags

Refine your search with specific, detailed tags. Use the same Include/Exclude logic.

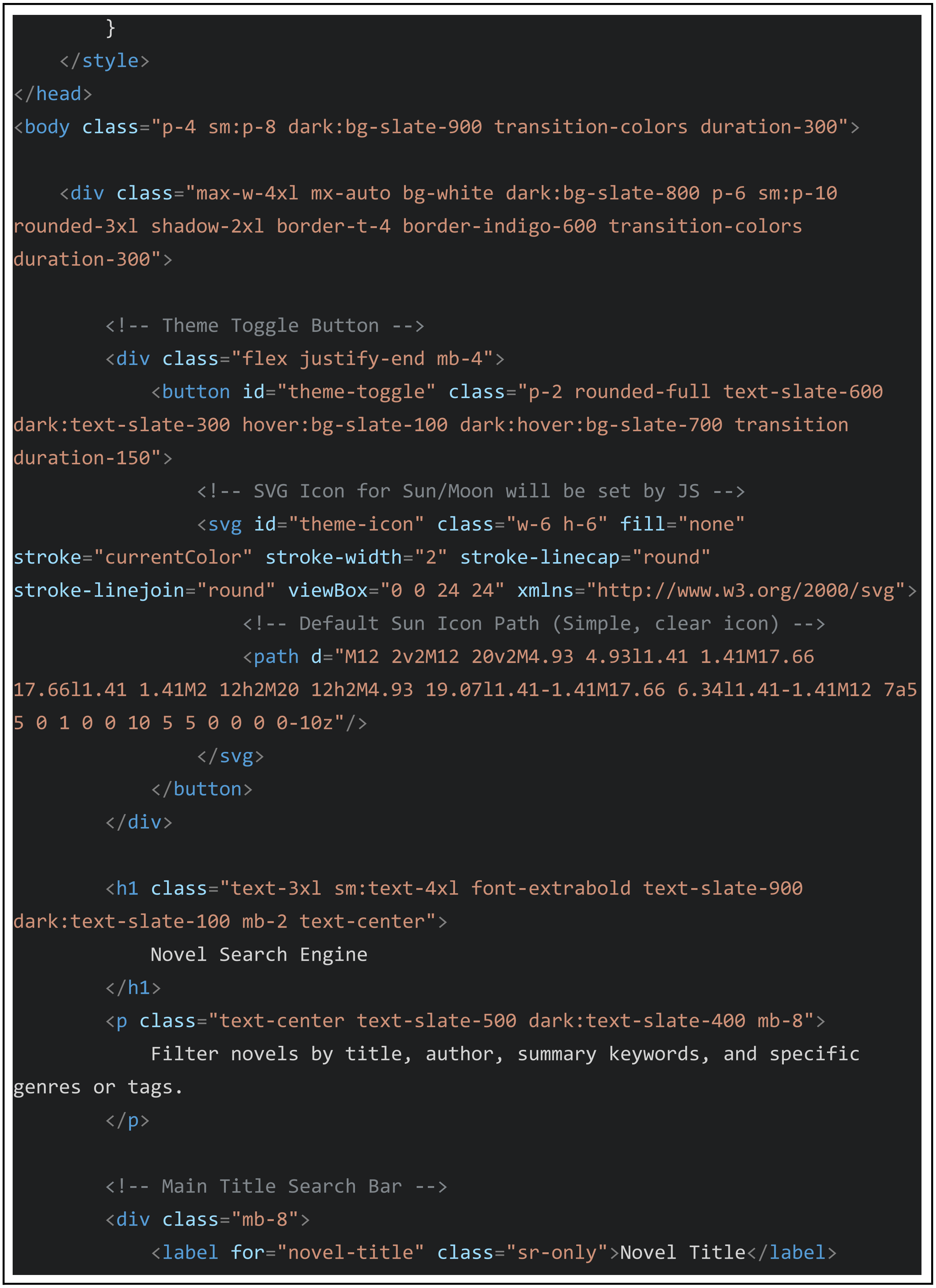
Villainess Time Travel Transmigration System VRMMO Slice of Life Dungeons Harem Rebirth

Code:

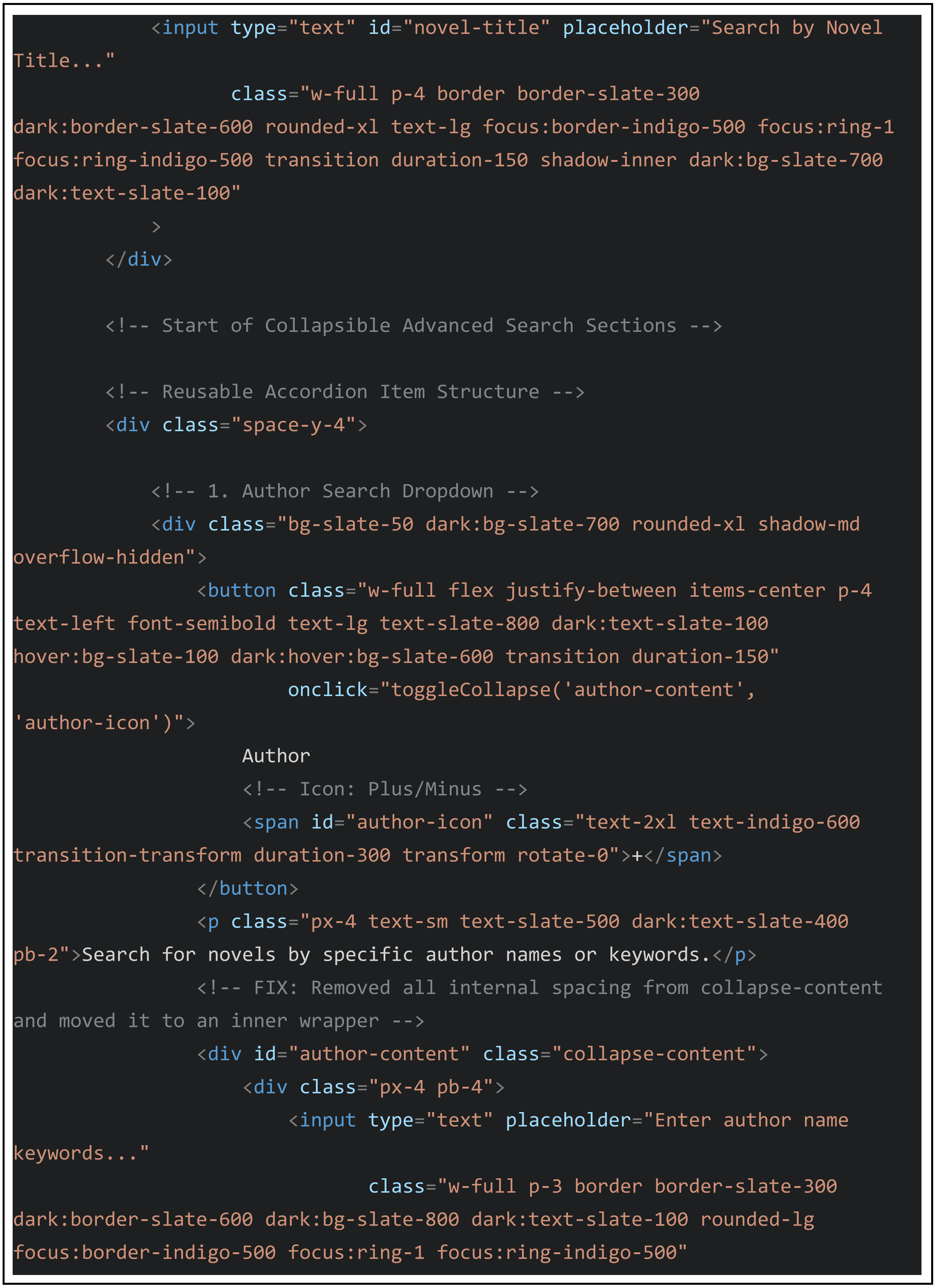
```
<!DOCTYPE html>
<html lang="en">
<head>
```



```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Advanced Novel Search Interface</title>
    <!-- Load Tailwind CSS and configure Dark Mode -->
    <script src="https://cdn.tailwindcss.com"></script>
    <script>
        // Configure Tailwind to use the 'dark' class on the HTML element for dark mode
        tailwind.config = {
            darkMode: 'class',
            theme: {
                extend: {
                    fontFamily: {
                        sans: ['Inter', 'sans-serif'],
                    },
                },
            },
        }
    </script>
    <!-- Use Inter font -->
    <style>
        @import url('https://fonts.googleapis.com/css2?family=Inter:wght@100..900&display=swap');
        body {
            font-family: 'Inter', sans-serif;
            background-color: #f1f5f9; /* Light Slate Background */
            transition: background-color 0.3s;
        }
        /* Custom styles for smoother transition on collapse */
        .collapse-content {
            max-height: 0;
            overflow: hidden;
            transition: max-height 0.4s ease-in-out;
        }
        .collapse-content.active {
            /* Set a height that is guaranteed to be taller than the content */
            max-height: 1000px;
```

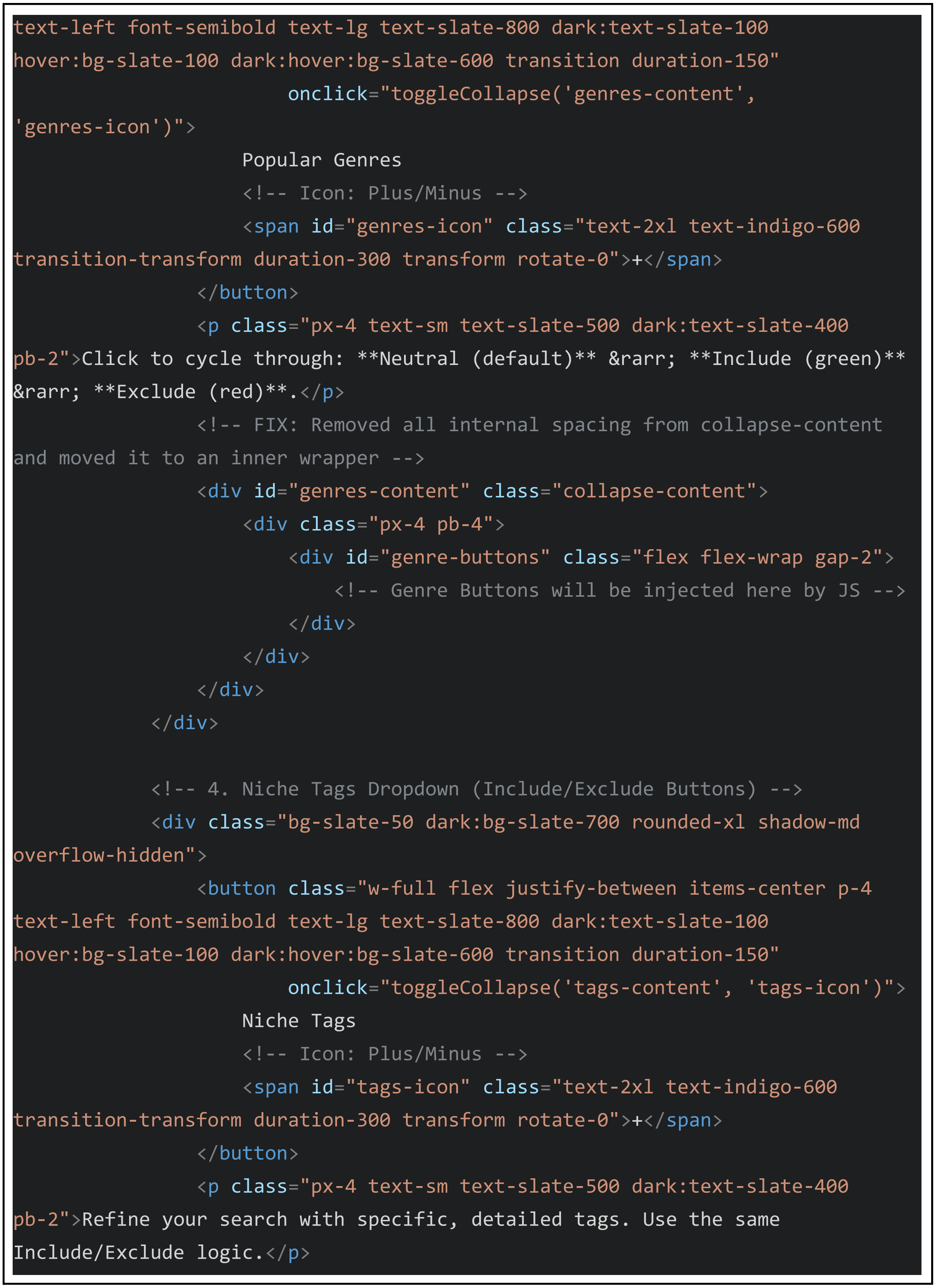


```html
        }
    </style>
</head>
<body class="p-4 sm:p-8 dark:bg-slate-900 transition-colors duration-300">

    <div class="max-w-4xl mx-auto bg-white dark:bg-slate-800 p-6 sm:p-10 rounded-3xl shadow-2xl border-t-4 border-indigo-600 transition-colors duration-300">

        <!-- Theme Toggle Button -->
        <div class="flex justify-end mb-4">
            <button id="theme-toggle" class="p-2 rounded-full text-slate-600 dark:text-slate-300 hover:bg-slate-100 dark:hover:bg-slate-700 transition duration-150">
                <!-- SVG Icon for Sun/Moon will be set by JS -->
                <svg id="theme-icon" class="w-6 h-6" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" viewBox="0 0 24 24" xmlns="http://www.w3.org/2000/svg">
                    <!-- Default Sun Icon Path (Simple, clear icon) -->
                    <path d="M12 2v2M12 20v2M4.93 4.93l1.41 1.41M17.66 17.66l1.41 1.41M2 12h2M20 12h2M4.93 19.07l1.41-1.41M17.66 6.34l1.41-1.41M12 7a5 5 0 1 0 0 10 5 5 0 0 0 0-10z"/>
                </svg>
            </button>
        </div>

        <h1 class="text-3xl sm:text-4xl font-extrabold text-slate-900 dark:text-slate-100 mb-2 text-center">
            Novel Search Engine
        </h1>
        <p class="text-center text-slate-500 dark:text-slate-400 mb-8">
            Filter novels by title, author, summary keywords, and specific genres or tags.
        </p>

        <!-- Main Title Search Bar -->
        <div class="mb-8">
            <label for="novel-title" class="sr-only">Novel Title</label>
```

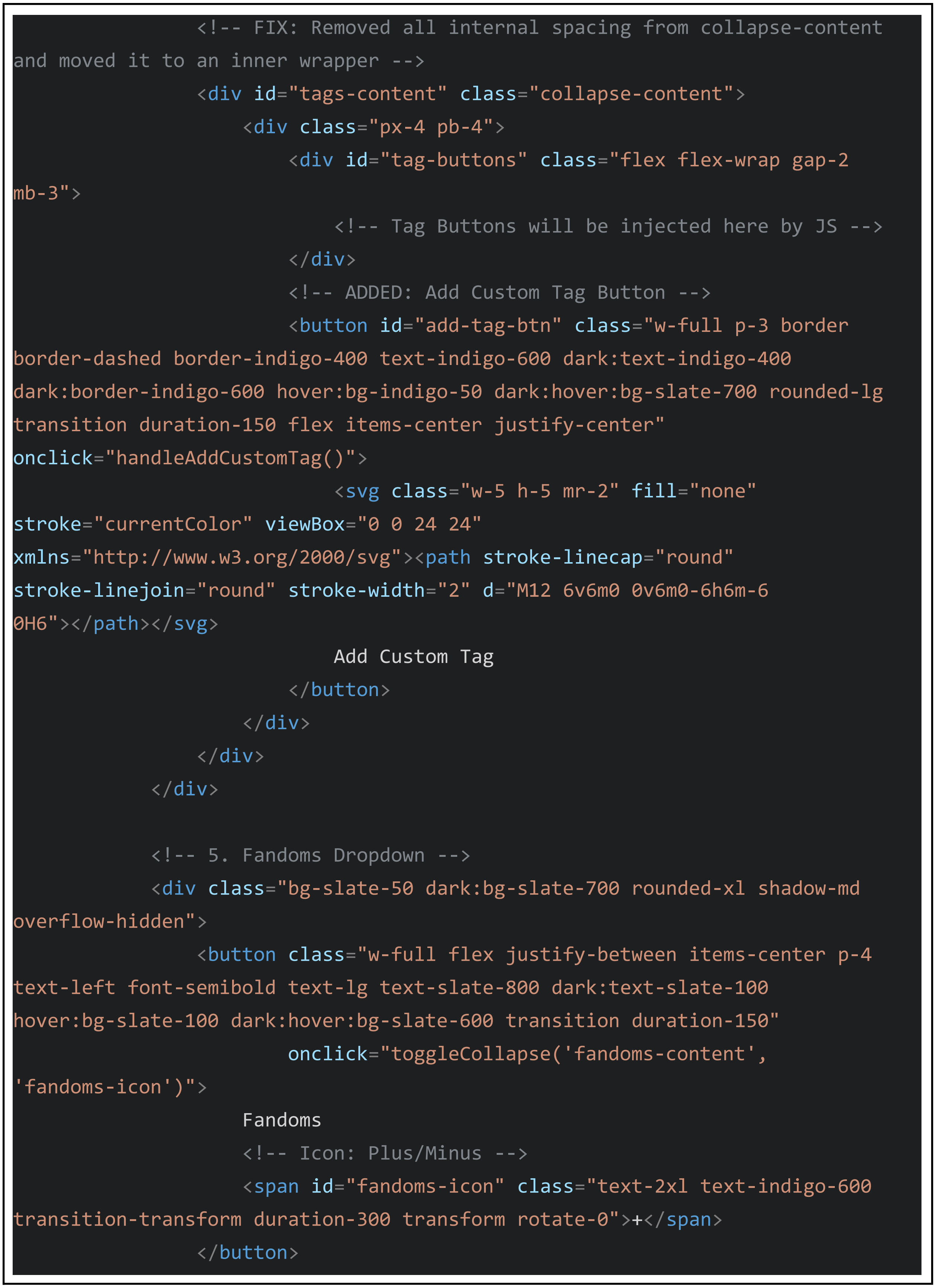


```
            <input type="text" id="novel-title" placeholder="Search by Novel
Title..."
                   class="w-full p-4 border border-slate-300
dark:border-slate-600 rounded-xl text-lg focus:border-indigo-500 focus:ring-1
focus:ring-indigo-500 transition duration-150 shadow-inner dark:bg-slate-700
dark:text-slate-100"
            >
        </div>

        <!-- Start of Collapsible Advanced Search Sections -->

        <!-- Reusable Accordion Item Structure -->
        <div class="space-y-4">

            <!-- 1. Author Search Dropdown -->
            <div class="bg-slate-50 dark:bg-slate-700 rounded-xl shadow-md
overflow-hidden">
                <button class="w-full flex justify-between items-center p-4
text-left font-semibold text-lg text-slate-800 dark:text-slate-100
hover:bg-slate-100 dark:hover:bg-slate-600 transition duration-150"
                        onclick="toggleCollapse('author-content',
'author-icon')">
                    Author
                    <!-- Icon: Plus/Minus -->
                    <span id="author-icon" class="text-2xl text-indigo-600
transition-transform duration-300 transform rotate-0">+</span>
                </button>
                <p class="px-4 text-sm text-slate-500 dark:text-slate-400
pb-2">Search for novels by specific author names or keywords.</p>
                <!-- FIX: Removed all internal spacing from collapse-content
and moved it to an inner wrapper -->
                <div id="author-content" class="collapse-content">
                    <div class="px-4 pb-4">
                        <input type="text" placeholder="Enter author name
keywords..."
                               class="w-full p-3 border border-slate-300
dark:border-slate-600 dark:bg-slate-800 dark:text-slate-100 rounded-lg
focus:border-indigo-500 focus:ring-1 focus:ring-indigo-500"
```

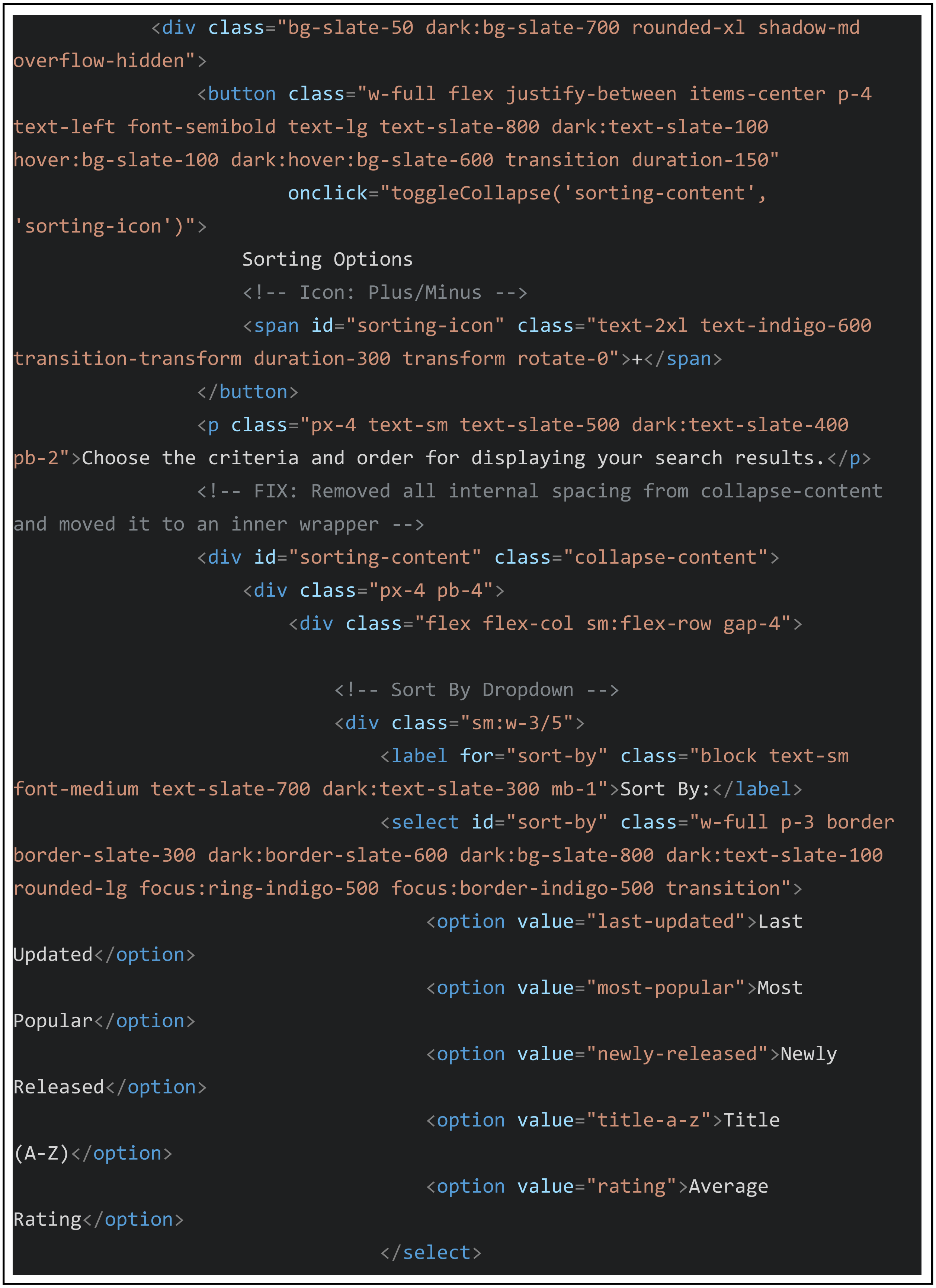
```
                        >
                    </div>
                </div>
            </div>

            <!-- 2. Summary Search Dropdown -->
            <div class="bg-slate-50 dark:bg-slate-700 rounded-xl shadow-md
overflow-hidden">
                <button class="w-full flex justify-between items-center p-4
text-left font-semibold text-lg text-slate-800 dark:text-slate-100
hover:bg-slate-100 dark:hover:bg-slate-600 transition duration-150"
                        onclick="toggleCollapse('summary-content',
'summary-icon')">
                    Summary Keywords
                    <!-- Icon: Plus/Minus -->
                    <span id="summary-icon" class="text-2xl text-indigo-600
transition-transform duration-300 transform rotate-0">+</span>
                </button>
                <p class="px-4 text-sm text-slate-500 dark:text-slate-400
pb-2">Use keywords found within the novel's summary description.</p>
                <!-- FIX: Removed all internal spacing from collapse-content
and moved it to an inner wrapper -->
                <div id="summary-content" class="collapse-content">
                    <div class="px-4 pb-4">
                        <input type="text" placeholder="Enter summary
keywords..."
                               class="w-full p-3 border border-slate-300
dark:border-slate-600 dark:bg-slate-800 dark:text-slate-100 rounded-lg
focus:border-indigo-500 focus:ring-1 focus:ring-indigo-500"
                        >
                    </div>
                </div>
            </div>

            <!-- 3. Popular Genres Dropdown (Include/Exclude Buttons) -->
            <div class="bg-slate-50 dark:bg-slate-700 rounded-xl shadow-md
overflow-hidden">
                <button class="w-full flex justify-between items-center p-4
```



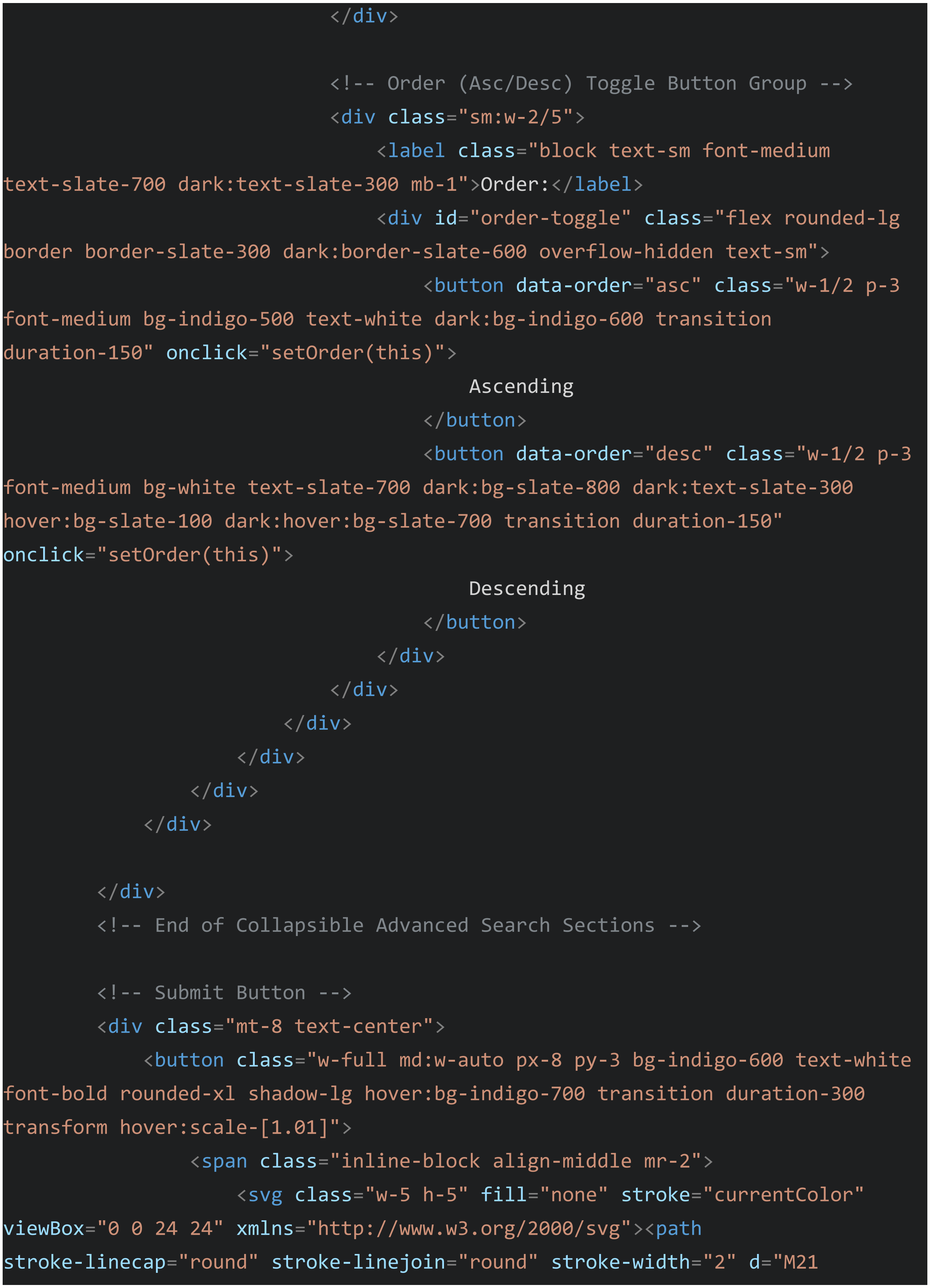
```
text-left font-semibold text-lg text-slate-800 dark:text-slate-100
hover:bg-slate-100 dark:hover:bg-slate-600 transition duration-150"
                        onclick="toggleCollapse('genres-content',
'genres-icon')">
                    Popular Genres
                    <!-- Icon: Plus/Minus -->
                    <span id="genres-icon" class="text-2xl text-indigo-600
transition-transform duration-300 transform rotate-0">+</span>
                </button>
                <p class="px-4 text-sm text-slate-500 dark:text-slate-400
pb-2">Click to cycle through: **Neutral (default)** &rarr; **Include (green)**
&rarr; **Exclude (red)**.</p>
                <!-- FIX: Removed all internal spacing from collapse-content
and moved it to an inner wrapper -->
                <div id="genres-content" class="collapse-content">
                    <div class="px-4 pb-4">
                        <div id="genre-buttons" class="flex flex-wrap gap-2">
                            <!-- Genre Buttons will be injected here by JS -->
                        </div>
                    </div>
                </div>
            </div>

            <!-- 4. Niche Tags Dropdown (Include/Exclude Buttons) -->
            <div class="bg-slate-50 dark:bg-slate-700 rounded-xl shadow-md
overflow-hidden">
                <button class="w-full flex justify-between items-center p-4
text-left font-semibold text-lg text-slate-800 dark:text-slate-100
hover:bg-slate-100 dark:hover:bg-slate-600 transition duration-150"
                        onclick="toggleCollapse('tags-content', 'tags-icon')">
                    Niche Tags
                    <!-- Icon: Plus/Minus -->
                    <span id="tags-icon" class="text-2xl text-indigo-600
transition-transform duration-300 transform rotate-0">+</span>
                </button>
                <p class="px-4 text-sm text-slate-500 dark:text-slate-400
pb-2">Refine your search with specific, detailed tags. Use the same
Include/Exclude logic.</p>
```



```
                <!-- FIX: Removed all internal spacing from collapse-content
and moved it to an inner wrapper -->
                <div id="tags-content" class="collapse-content">
                    <div class="px-4 pb-4">
                        <div id="tag-buttons" class="flex flex-wrap gap-2
mb-3">
                            <!-- Tag Buttons will be injected here by JS -->
                        </div>
                        <!-- ADDED: Add Custom Tag Button -->
                        <button id="add-tag-btn" class="w-full p-3 border
border-dashed border-indigo-400 text-indigo-600 dark:text-indigo-400
dark:border-indigo-600 hover:bg-indigo-50 dark:hover:bg-slate-700 rounded-lg
transition duration-150 flex items-center justify-center"
onclick="handleAddCustomTag()">
                            <svg class="w-5 h-5 mr-2" fill="none"
stroke="currentColor" viewBox="0 0 24 24"
xmlns="http://www.w3.org/2000/svg"><path stroke-linecap="round"
stroke-linejoin="round" stroke-width="2" d="M12 6v6m0 0v6m0-6h6m-6
0H6"></path></svg>
                            Add Custom Tag
                        </button>
                    </div>
                </div>
            </div>

            <!-- 5. Fandoms Dropdown -->
            <div class="bg-slate-50 dark:bg-slate-700 rounded-xl shadow-md
overflow-hidden">
                <button class="w-full flex justify-between items-center p-4
text-left font-semibold text-lg text-slate-800 dark:text-slate-100
hover:bg-slate-100 dark:hover:bg-slate-600 transition duration-150"
                        onclick="toggleCollapse('fandoms-content',
'fandoms-icon')">
                    Fandoms
                    <!-- Icon: Plus/Minus -->
                    <span id="fandoms-icon" class="text-2xl text-indigo-600
transition-transform duration-300 transform rotate-0">+</span>
                </button>
                <p class="px-4 text-sm text-slate-500 dark:text-slate-400
pb-2">Filter results related to specific popular fictional universes (shows,
movies, comics, etc.).</p>
                <div id="fandoms-content" class="collapse-content">
                    <div class="px-4 pb-4">
                        <!-- Search Input -->
                        <input type="text" id="fandom-search"
placeholder="Quick find a fandom..."
                               class="w-full p-3 mb-3 border border-slate-300
dark:border-slate-600 dark:bg-slate-800 dark:text-slate-100 rounded-lg
focus:border-indigo-500 focus:ring-1 focus:ring-indigo-500"
                        >

                        <!-- Fandom Buttons -->
                        <div id="fandom-buttons" class="flex flex-wrap gap-2
mb-3">
                            <!-- Fandom Buttons will be injected here by JS -->
                        </div>

                        <!-- Add Custom Fandom Button -->
                        <button id="add-fandom-btn" class="w-full p-3 border
border-dashed border-indigo-400 text-indigo-600 dark:text-indigo-400
dark:border-indigo-600 hover:bg-indigo-50 dark:hover:bg-slate-700 rounded-lg
transition duration-150 flex items-center justify-center"
onclick="handleAddCustomFandom()">
                            <svg class="w-5 h-5 mr-2" fill="none"
stroke="currentColor" viewBox="0 0 24 24"
xmlns="http://www.w3.org/2000/svg"><path stroke-linecap="round"
stroke-linejoin="round" stroke-width="2" d="M12 6v6m0 0v6m0-6h6m-6
0H6"></path></svg>
                            Add Custom Fandom
                        </button>

                    </div>
                </div>
            </div>

            <!-- 6. Sorting Options Dropdown -->
```

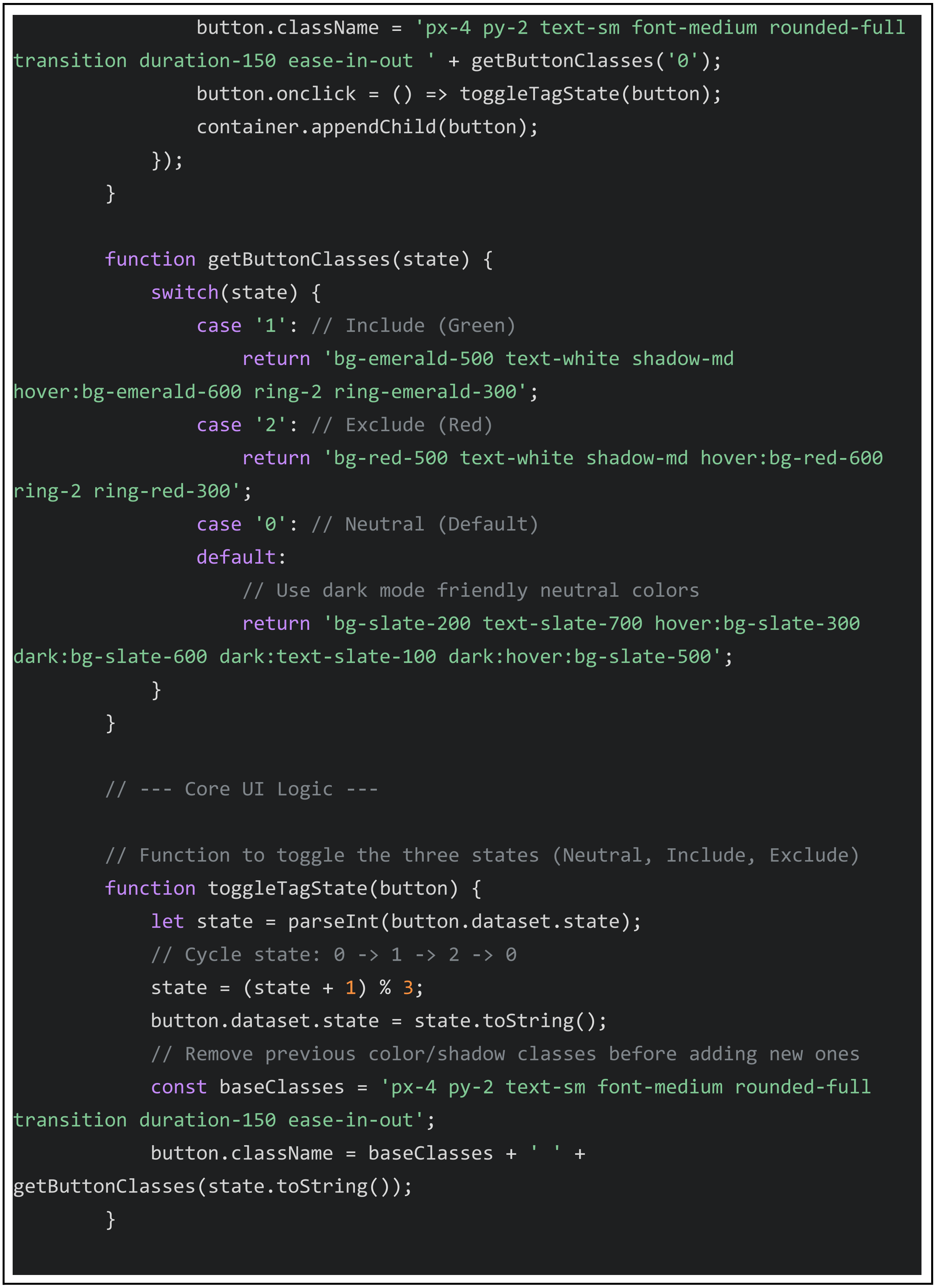


```
            <div class="bg-slate-50 dark:bg-slate-700 rounded-xl shadow-md overflow-hidden">
                <button class="w-full flex justify-between items-center p-4 text-left font-semibold text-lg text-slate-800 dark:text-slate-100 hover:bg-slate-100 dark:hover:bg-slate-600 transition duration-150"
                        onclick="toggleCollapse('sorting-content', 'sorting-icon')">
                    Sorting Options
                    <!-- Icon: Plus/Minus -->
                    <span id="sorting-icon" class="text-2xl text-indigo-600 transition-transform duration-300 transform rotate-0">+</span>
                </button>
                <p class="px-4 text-sm text-slate-500 dark:text-slate-400 pb-2">Choose the criteria and order for displaying your search results.</p>
                <!-- FIX: Removed all internal spacing from collapse-content and moved it to an inner wrapper -->
                <div id="sorting-content" class="collapse-content">
                    <div class="px-4 pb-4">
                        <div class="flex flex-col sm:flex-row gap-4">

                            <!-- Sort By Dropdown -->
                            <div class="sm:w-3/5">
                                <label for="sort-by" class="block text-sm font-medium text-slate-700 dark:text-slate-300 mb-1">Sort By:</label>
                                <select id="sort-by" class="w-full p-3 border border-slate-300 dark:border-slate-600 dark:bg-slate-800 dark:text-slate-100 rounded-lg focus:ring-indigo-500 focus:border-indigo-500 transition">
                                    <option value="last-updated">Last Updated</option>
                                    <option value="most-popular">Most Popular</option>
                                    <option value="newly-released">Newly Released</option>
                                    <option value="title-a-z">Title (A-Z)</option>
                                    <option value="rating">Average Rating</option>
                                </select>
```

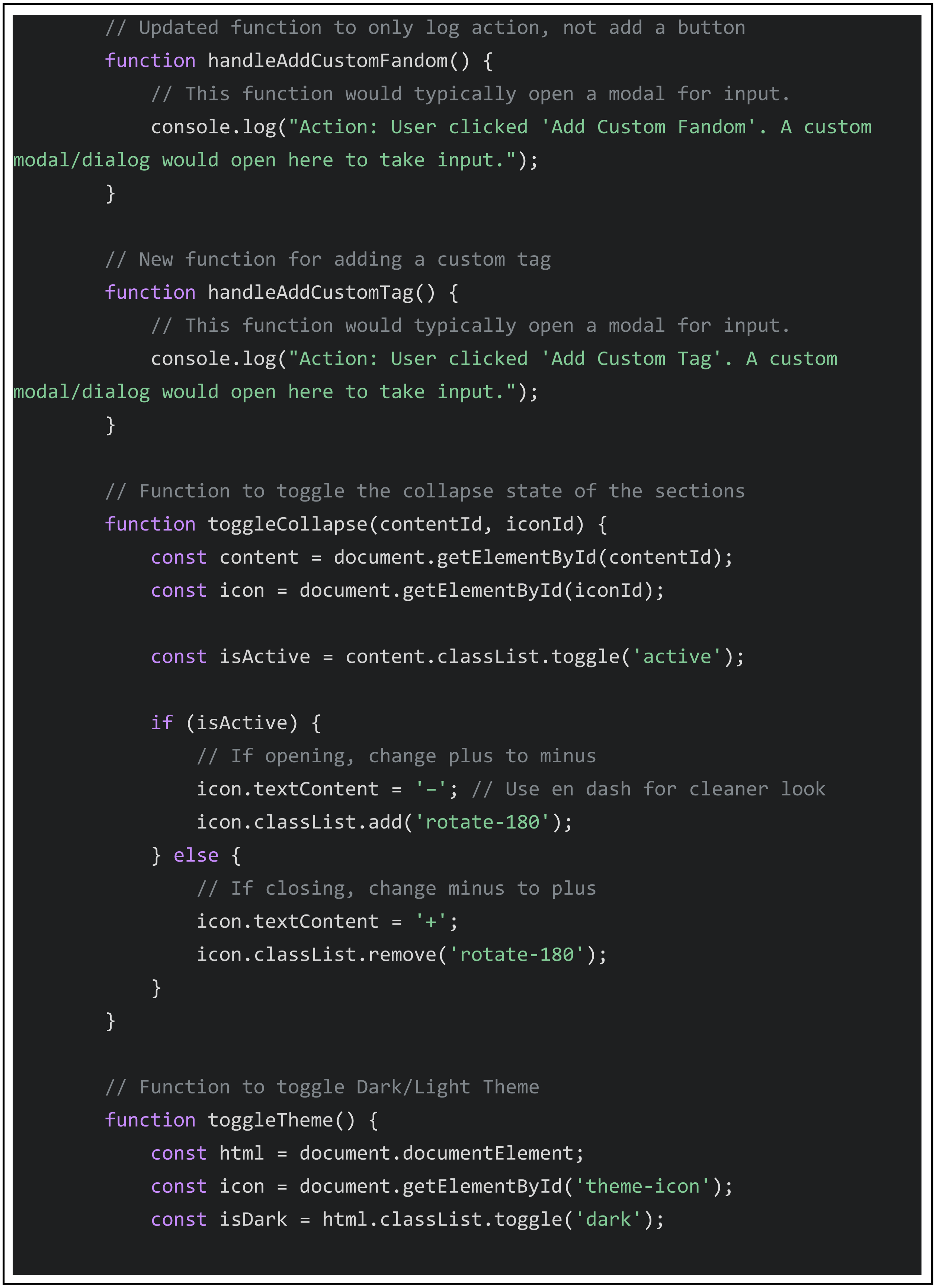


```
                            </div>

                            <!-- Order (Asc/Desc) Toggle Button Group -->
                            <div class="sm:w-2/5">
                                <label class="block text-sm font-medium
text-slate-700 dark:text-slate-300 mb-1">Order:</label>
                                <div id="order-toggle" class="flex rounded-lg
border border-slate-300 dark:border-slate-600 overflow-hidden text-sm">
                                    <button data-order="asc" class="w-1/2 p-3
font-medium bg-indigo-500 text-white dark:bg-indigo-600 transition
duration-150" onclick="setOrder(this)">
                                        Ascending
                                    </button>
                                    <button data-order="desc" class="w-1/2 p-3
font-medium bg-white text-slate-700 dark:bg-slate-800 dark:text-slate-300
hover:bg-slate-100 dark:hover:bg-slate-700 transition duration-150"
onclick="setOrder(this)">
                                        Descending
                                    </button>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>

        </div>
        <!-- End of Collapsible Advanced Search Sections -->

        <!-- Submit Button -->
        <div class="mt-8 text-center">
            <button class="w-full md:w-auto px-8 py-3 bg-indigo-600 text-white
font-bold rounded-xl shadow-lg hover:bg-indigo-700 transition duration-300
transform hover:scale-[1.01]">
                <span class="inline-block align-middle mr-2">
                    <svg class="w-5 h-5" fill="none" stroke="currentColor"
viewBox="0 0 24 24" xmlns="http://www.w3.org/2000/svg"><path
stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M21
```

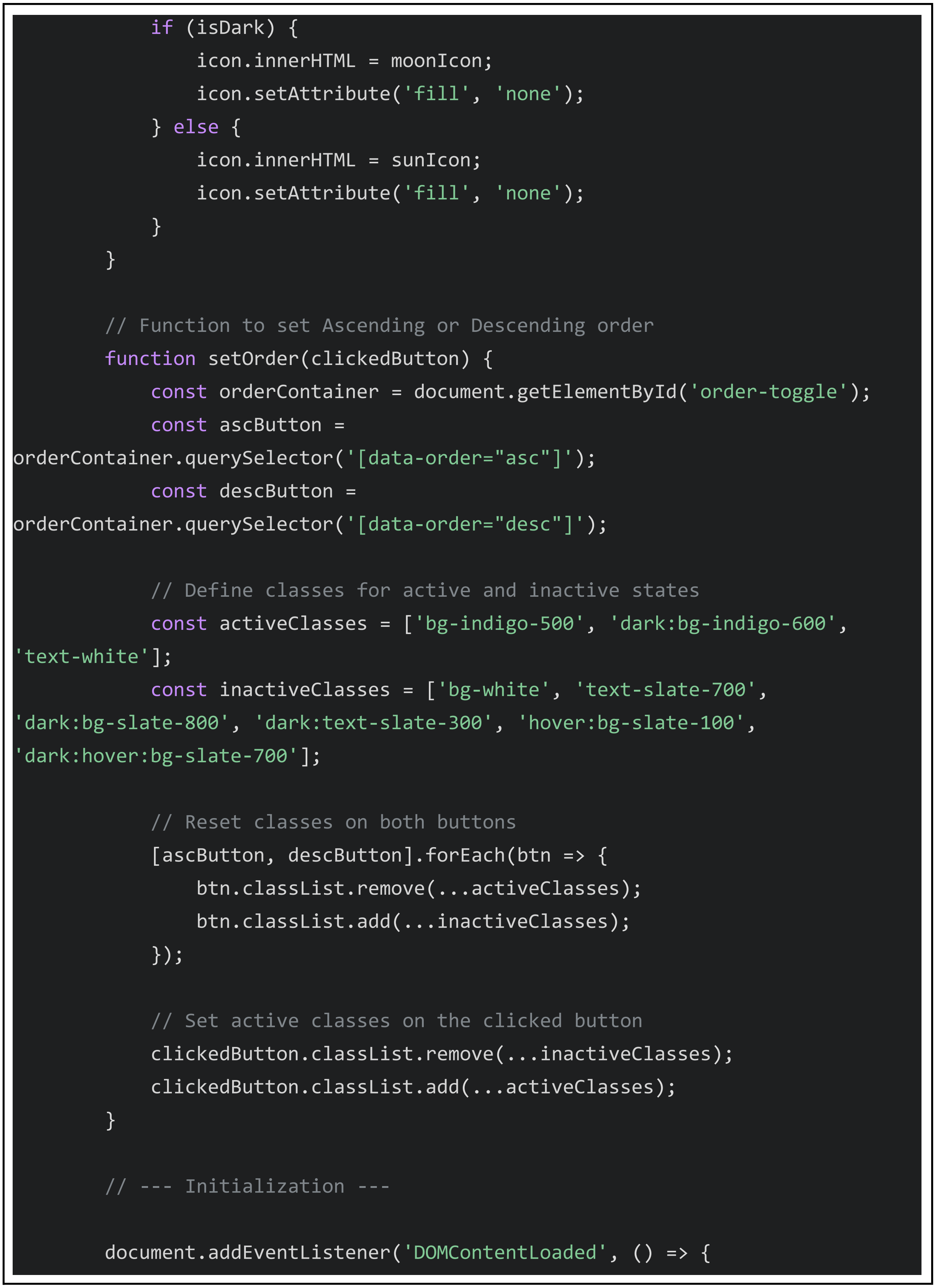
```
21l-6-6m2-5a7 7 0 11-14 0 7 7 0 0114 0z"></path></svg>
                </span>
                Run Advanced Search
            </button>
        </div>

    </div>

    <script>
        const GENRES = ['Fantasy', 'Romance', 'Sci-Fi', 'Action', 'Mystery', 'Historical', 'Cultivation', 'Gaming', 'Adventure'];
        const TAGS = ['Villainess', 'Time Travel', 'Transmigration', 'System', 'VRMMO', 'Slice of Life', 'Dungeons', 'Harem', 'Rebirth'];
        // New Fandoms Data
        const FANDOMS = ['Star Wars', 'Harry Potter', 'Marvel MCU', 'DC Comics', 'Doctor Who', 'Lord of the Rings', 'Game of Thrones', 'Avatar: The Last Airbender'];

        // Updated SVG path data for cleaner icons (using stroke for clarity)
        const sunIcon = `<path d="M12 2v2M12 20v2M4.93 4.93l1.41 1.41M17.66 17.66l1.41 1.41M2 12h2M20 12h2M4.93 19.07l1.41-1.41M17.66 6.34l1.41-1.41M12 7a5 5 0 1 0 0 10 5 5 0 0 0 0-10z"/>`;
        const moonIcon = `<path d="M21 12.79A9 9 0 1 1 11.21 3 7 7 0 0 0 21 12.79z"/>`; // A clear crescent moon

        // --- Data and Initialization ---

        function createButtons(containerId, items) {
            const container = document.getElementById(containerId);
            // Clear existing buttons first, in case of dynamic additions
            container.innerHTML = '';

            items.forEach(item => {
                const button = document.createElement('button');
                button.textContent = item;
                button.dataset.state = '0';
                button.dataset.tagType = containerId; // e.g., 'genre-buttons', 'tag-buttons', 'fandom-buttons'
```



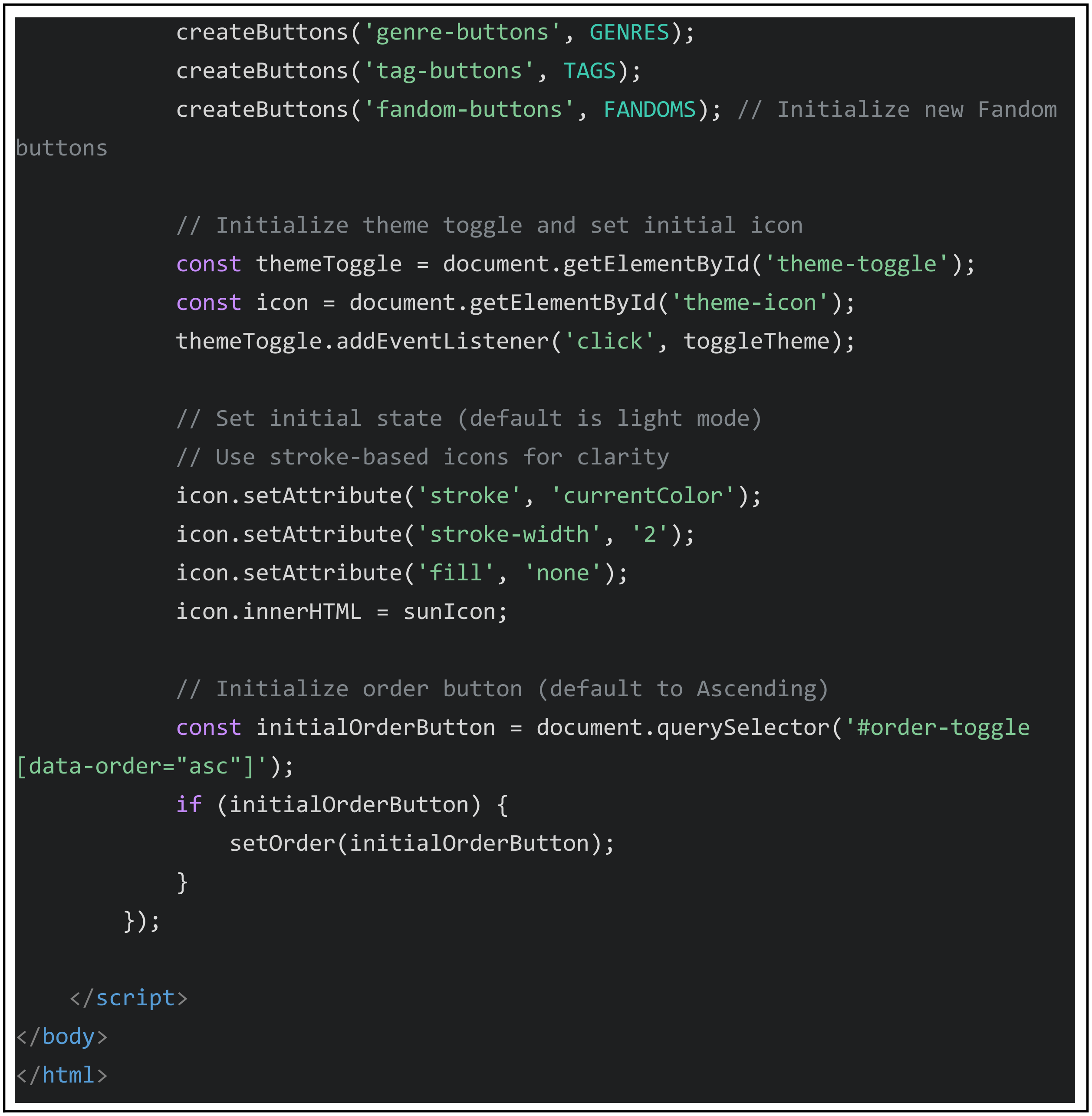
```
                button.className = 'px-4 py-2 text-sm font-medium rounded-full 
transition duration-150 ease-in-out ' + getButtonClasses('0');
                button.onclick = () => toggleTagState(button);
                container.appendChild(button);
            });
        }

        function getButtonClasses(state) {
            switch(state) {
                case '1': // Include (Green)
                    return 'bg-emerald-500 text-white shadow-md 
hover:bg-emerald-600 ring-2 ring-emerald-300';
                case '2': // Exclude (Red)
                    return 'bg-red-500 text-white shadow-md hover:bg-red-600 
ring-2 ring-red-300';
                case '0': // Neutral (Default)
                default:
                    // Use dark mode friendly neutral colors
                    return 'bg-slate-200 text-slate-700 hover:bg-slate-300 
dark:bg-slate-600 dark:text-slate-100 dark:hover:bg-slate-500';
            }
        }

        // --- Core UI Logic ---

        // Function to toggle the three states (Neutral, Include, Exclude)
        function toggleTagState(button) {
            let state = parseInt(button.dataset.state);
            // Cycle state: 0 -> 1 -> 2 -> 0
            state = (state + 1) % 3;
            button.dataset.state = state.toString();
            // Remove previous color/shadow classes before adding new ones
            const baseClasses = 'px-4 py-2 text-sm font-medium rounded-full 
transition duration-150 ease-in-out';
            button.className = baseClasses + ' ' + 
getButtonClasses(state.toString());
        }
```



```javascript
        // Updated function to only log action, not add a button
        function handleAddCustomFandom() {
            // This function would typically open a modal for input.
            console.log("Action: User clicked 'Add Custom Fandom'. A custom modal/dialog would open here to take input.");
        }

        // New function for adding a custom tag
        function handleAddCustomTag() {
            // This function would typically open a modal for input.
            console.log("Action: User clicked 'Add Custom Tag'. A custom modal/dialog would open here to take input.");
        }

        // Function to toggle the collapse state of the sections
        function toggleCollapse(contentId, iconId) {
            const content = document.getElementById(contentId);
            const icon = document.getElementById(iconId);

            const isActive = content.classList.toggle('active');

            if (isActive) {
                // If opening, change plus to minus
                icon.textContent = '–'; // Use en dash for cleaner look
                icon.classList.add('rotate-180');
            } else {
                // If closing, change minus to plus
                icon.textContent = '+';
                icon.classList.remove('rotate-180');
            }
        }

        // Function to toggle Dark/Light Theme
        function toggleTheme() {
            const html = document.documentElement;
            const icon = document.getElementById('theme-icon');
            const isDark = html.classList.toggle('dark');
```



```
            if (isDark) {
                icon.innerHTML = moonIcon;
                icon.setAttribute('fill', 'none');
            } else {
                icon.innerHTML = sunIcon;
                icon.setAttribute('fill', 'none');
            }
        }

        // Function to set Ascending or Descending order
        function setOrder(clickedButton) {
            const orderContainer = document.getElementById('order-toggle');
            const ascButton =
orderContainer.querySelector('[data-order="asc"]');
            const descButton =
orderContainer.querySelector('[data-order="desc"]');

            // Define classes for active and inactive states
            const activeClasses = ['bg-indigo-500', 'dark:bg-indigo-600',
'text-white'];
            const inactiveClasses = ['bg-white', 'text-slate-700',
'dark:bg-slate-800', 'dark:text-slate-300', 'hover:bg-slate-100',
'dark:hover:bg-slate-700'];

            // Reset classes on both buttons
            [ascButton, descButton].forEach(btn => {
                btn.classList.remove(...activeClasses);
                btn.classList.add(...inactiveClasses);
            });

            // Set active classes on the clicked button
            clickedButton.classList.remove(...inactiveClasses);
            clickedButton.classList.add(...activeClasses);
        }

        // --- Initialization ---

        document.addEventListener('DOMContentLoaded', () => {
```



```
            createButtons('genre-buttons', GENRES);
            createButtons('tag-buttons', TAGS);
            createButtons('fandom-buttons', FANDOMS); // Initialize new Fandom
buttons

            // Initialize theme toggle and set initial icon
            const themeToggle = document.getElementById('theme-toggle');
            const icon = document.getElementById('theme-icon');
            themeToggle.addEventListener('click', toggleTheme);

            // Set initial state (default is light mode)
            // Use stroke-based icons for clarity
            icon.setAttribute('stroke', 'currentColor');
            icon.setAttribute('stroke-width', '2');
            icon.setAttribute('fill', 'none');
            icon.innerHTML = sunIcon;

            // Initialize order button (default to Ascending)
            const initialOrderButton = document.querySelector('#order-toggle
[data-order="asc"]');
            if (initialOrderButton) {
                setOrder(initialOrderButton);
            }
        });

    </script>
</body>
</html>
```