Music Manager for Plex

Music Manager for Plex is a powerful Streamlit application running on Docker, locally on a PC, or Unraid to organize, automate, and curate your Plex Music library.

Key Features:

- **Export Metadata:** Dump your entire library (Tracks, Albums, Artists) to CSV.
- **Smart Playlist Creator:** Generate "Sonic" playlists using Plex's sonic analysis, listening history, and custom seeds (e.g., "Jazz Mix" or "Radiohead Radio").
- **Bulk Editor:** Update metadata via CSV (Genres, Ratings, Dates, Collections) using single or chained scripts.
- **Metadata Comparator:** Compare two export files to see exactly what changed in your library over time.

Table of Contents

1. Installation

1.1 Local Installation

Run the application locally on any Windows, Mac, or Linux machine with Python installed.

Prerequisites:

- Python 3.8 or higher: Download Python (Make sure to check "Add Python to PATH" during installation).

Steps:

1. Download the application:
   - Clone the repository: git clone https://github.com/caddickzac/Music-Manager-for-Plex.git
   - OR Download the ZIP file and extract it to a folder.
   - Open a terminal/command prompt and navigate to the application folder:

   Python Syntax:
   ```
   cd path/to/Music-Manager-for-Plex
   ```

2. Install dependencies:

   Python Syntax:
   ```
   pip install -r requirements.txt
   ```

3. Run the application:

   Python Syntax:
   ```
   streamlit run Music-Manager-for-Plex-App.py
   ```

4. Access the Interface:
   - The application should automatically open in your default web browser.
   - If not, visit: http://localhost:8501

Configuration (Optional): You can create a config.txt file in the root directory to save your Plex URL and Token so you don't have to enter them every time. (See config_example.txt if included).

1.2 Unraid Version

The application is available directly through the Unraid Community Applications (CA) store.

Steps:

1. Open Unraid WebUI and navigate to the Apps tab.
2. Search for "Music-Manager-for-Plex".
3. Click Install.
4. Configure Container Settings:
   - Port: Ensure the WebUI port (default 8501) does not conflict with other services.
   - Add Path variables
     - App Data: Add an App Data folder for metadata saving to your Unraid array (e.g., /mnt/user/appdata/Music-Manager-for-Plex/)
     - Exports: Add an Exports folder for metadata saving to your Unraid array (e.g., /mnt/user/appdata/Music-Manager-for-Plex/exports).

        - *Important:* If you do not map this volume, your exported CSV files will be lost if the container restarts.
    - Environment Variables (Optional): You can pre-fill your connection details by adding these variables:
        - PLEX_URL: Your Plex Server URL (e.g., http://192.168.1.5:32400)
        - PLEX_TOKEN: Your Plex Authentication Token.
        - Optional: APP_PASSWORD: **********
5. Click Apply to start the container.
6. Open WebUI: Click on the container icon and select "WebUI" to launch the tool.

1.3 Docker Version

The latest stable image is available on Docker Hub.
You can pull it directly using the following command:

```
Python Syntax:
docker pull zcaddick/music-manager-for-plex:latest
```

2. Configuration

2.1 Local Configuration

Create a file named config.txt in the root folder.

Plex URL: e.g., http://192.168.1.50:32400
Plex Token: abcXYZ123token
Plex Library: Music

Note: Add the txt file to your .gitignore to keep your token safe!

2.2 Unraid & Docker Configuration

Edit Environment Variables in Unraid Docker App.

The app needs these variable:

- PLEX_URL
- PLEX_TOKEN
- PLEX_LIBRARY_NAME (e.g., "Music", "HiFi", "FLAC")
- In Docker you need to add a Port path too.

3. Export

The **Export** module allows you to extract metadata from your Plex library into CSV files. This is useful for backing up your library data, analyzing your listening habits in Excel, or preparing data for mass updates.

- **Functionality:** Connects to your Plex server and scans the selected music library.
- **Export Levels:**
    - **Tracks:** Exports individual song data (e.g., title, rating, play count, skip count, date added, playlist/collection information).
    - **Albums:** Exports album-level data (e.g., album title, year, track counts, average bitrate).

- **Artists:** Exports artist-level data (e.g., album lists for each artist, track counts, collection information, total file size).
- **Output:** Generates timestamped CSV files in the Exports folder. These files serve as the template for the Update modules.

Note: See the **Music_Manager_Track_Level_Data_Dictionary** file for descriptions of the exported metadata fields.

4. Update from CSV (Single Script)

This tool allows you to mass-edit your Plex metadata by uploading a modified CSV file. It is an efficient way to perform bulk changes (e.g., re-tagging genres, manually setting ratings, or fixing track titles).

Workflow:

1. **Export**: First export track details from the **Export** tab. This provides a detailed CSV file of your music library at three levels of granularity:
    - track (“YY_MM_DD **Track_Level_Info**.csv”)
    - album (“YY_MM_DD **Artist_Album_Info**.csv”)
    - artist (“YY_MM_DD **Artist_Level_Info**.csv”)
2. **Edit:** Make your edits to the track level csv, this is the file you will upload. Tip: Just copy the rows you are editing to a new file. Remember to include the file header row.
3. **Variable Mapping:** The tool matches rows to your Plex content using unique track IDs (ratingKey). The tool is looking for specific variable names based on the specific action you choose. Check out the **Expected CSV schema & values** dropdown box for information for each action type.
4. **Choose an Action for Your Update:** Select your action from the **Choose an action** list.
5. **Upload csv with your edits:** Either drag and drop or look up your file on the **Drag and drop file here** button.
6. **Confirm action:** Type *CONFIRM* to sign off on your submission. Warning: Edits to metadata in plex can cause hard-to-fix errors or even corrupt your database. Make sure to make a backup of your data first and complete small test runs (3-5 tracks) before committing an action.
7. **Run:** Commits the changes to your Plex Media Server.

Notes:

- Editing track genres is computationally taxing because each track’s genre metadata must be ‘locked’ after updating. Be mindful when using this action and check computation times with small batches (50 tracks or less) before committing to a longer run.
- Editing artist titles is not possible via the API. This must be manually changed in the Plex app.

5. Update from CSV (Multiple Scripts)

This tool works like the Update from CSV (Single Script), but it allows for multiple scripts to run one after another. For example, you can edit an artists’ genre, album genre, and track genre information in one run. The same rules for variable mapping apply for each action. See section 2. Update from CSV (Single Script) for more details.

6. Compare Exported Metadata

This analysis tool takes two export files (e.g., "Export_Jan_01.csv" and "Export_Feb_01.csv") and highlights the differences.

- Purpose:
  - Track how your library has changed across time.
  - Verify that a bulk update script worked correctly.
  - Identify accidental metadata changes or lost ratings.
- Output: Features a dynamic column selector allowing you to choose exactly which variables to check (e.g., Bitrate, Mood, Rating). Generates a detailed report of differences side-by-side.

7. Playlist Creator

The Playlist Creator is an advanced music control room for creating playlists in your Plex music library. Unlike standard Plex smart playlists which are rigid (e.g., "All Rock tracks"), this tool uses distinct seed modes and playlist parameters like Exploit/Explore logic to mix your favorite hits with lesser known tracks, balance them sonically, and allows for constraints like album release year, the added date to your library, or specifying acceptable genres.

7.1 Seed Strategy

Seed Strategy has different modes which are the primary strategy used to gather candidate tracks for a playlist.

7.1.1 Auto (infer from seeds/history)

Brief description: Intelligently guesses the best strategy based on what inputs you provide.

How it works:

- If Genre Seeds are present -> Defaults to Genre Mode.
- If Seed Tracks are present -> Defaults to Sonic Tracks Mix.
- If Seed Artists are present -> Defaults to Sonic Artist Mix.
- If no seeds are provided -> Defaults to History + Seeds (pure history).

7.1.2 Deep Dive (Seed Albums)

Brief description: Picks full albums related to your seed tracks (great for rediscovering deep cuts).

How it works:

- Harvest Seeds: Collects seed tracks from inputs.
- Logic*:* If Seed Artists are provided, it works harder, grabbing up to 15 seed tracks per artist to ensure it touches multiple albums.
- Identify Parent Albums: Looks up the Album ID for every seed track.
- Fetch Siblings: Grabs all valid tracks from those identified albums.
- Smart Sort: Sorts tracks on those albums based on Explore/Exploit (Popularity) and Recency Bias.
- Fair Share Distribution: Forces a "Fair Share" loop so no single album dominates.

Inputs it uses:

- Seed Artists: Uses "Deep Dive Target" to pull a wide sample.
- Seed Tracks/Playlists/Collections: Used to identify albums.
- Explore vs. Exploit: Determines if you get "hits" or "deep cuts" from the album.
- Exclude Played (Days): Crucial for "finishing" albums you started.

Inputs it ignores:

- Sonic Similarity: Ignored (uses Metadata, not Audio).
- Sonic Journey: Ignored.

7.1.3 Genre seeds

Brief description: Pulls top tracks from specific genres (e.g., "Shoegaze").

How It Works:

- Track-Level Search: Asks Plex for tracks tagged with the genre.
- Album-Level Fallback: If track search yields few results, searches for Albums tagged with the genre.
- Smart Sort: Ranks the pool using Explore/Exploit (Popularity).
- Strictness Check: If "Genre Strict" is on, verifies selected tracks match the genre tags (Track > Album > Artist).

Inputs It Uses:

- Genre Seeds: The engine of this mode.
- Explore vs. Exploit: Determines "Classics" vs "Deep Cuts".
- Genre Strictness: Gatekeeper for accuracy.

Inputs It Ignores:

- Sonic Similarity: Ignored.
- Seed Tracks/Artists: Ignored (focuses purely on the Genre list).

7.1.4 History + Seeds (Union)

Brief description: Combines tracks from your history (favorites) with any specific manual seeds you enter.

How it works:

- Creates a pool of your "Top Tracks" from the last X days (defined by History Lookback).
- "Unpacks" any Seed Playlists/Collections into individual tracks and adds them to the pool.
- Applies filters and sorts by popularity/bias.

Inputs It Uses:

- History Lookback: Determines the time window for history.
- Seed Tracks/Playlists/Collections: Added directly to the mix.

Inputs It Ignores:

- Sonic Similarity: No sonic matching is performed; tracks are added raw.

7.1.5 Sonic Artist Mix

Brief description: Finds artists sonically similar to your seeds and picks their top tracks.

How it works:

- Takes your Seed Artists (or the artists of your Seed Tracks).
- Asks Plex for the "Sonically Similar Artists" for each.
- Grabs the top tracks from those *related* artists.

Inputs It Uses:

- Seed Artists: Primary input.
- Sonically Similar per Seed: Determines how many related artists to find.

7.1.6 Sonic Album Mix

Brief description: Finds full albums sonically similar to your seed albums.

How it works:

- Identifies the parent albums of your Seed Tracks.
- Asks Plex for "Sonically Similar Albums."
- Picks tracks from those *related* albums (applying Explore/Exploit logic).

Inputs It Uses:

- Seed Tracks/Albums: Used to find the source albums.
- Sonically Similar per Seed: Determines how many related albums to fetch.

7.1.7 Sonic Tracks Mix

Brief description: Finds individual tracks that match the sonic profile of your seed tracks.

How it works:

- Takes every Seed Track (unpacked from playlists/collections).
- Asks Plex for the closest "Sonically Similar Tracks."
- Creates a pool of these lookalikes.

Inputs It Uses:

- Seed Tracks/Playlists/Collections: The source targets.
- Sonically Similar per Seed: Controls match precision (Lower = Tighter match, Higher = More variety).

7.1.8 Sonic Combo

Brief description: A blend of Artist, Album, and Track sonic matching for maximum variety.

How it works:

- Runs Sonic Artist, Sonic Album, and Sonic Track logic simultaneously.
- Merges the results into a massive candidate pool.
- Ideal for filling large playlists where you want variety in how the music relates (some sound alike, some are from similar artists).

Inputs It Uses:

- All Seed Inputs: Artists, Tracks, and Albums are all utilized.

7.1.9 Sonic History (Intersection)

Brief description: Finds tracks from your history that *also* match the sonic profile of your seeds (e.g., "Songs I own and love that sound like Tame Impala”).

How it works:

- Pool A: Harvests your listening history.

- Pool B: Performs a Sonic Track search based on your seeds.
- Intersection: Keeps ONLY tracks that appear in both pools.

Inputs It Uses:

- History Lookback: Defines the "History" pool.
- Seed Tracks: Defines the "Sonic" pool.

Best Used For: Rediscovering favorites with a specific vibe.

7.1.10 Sonic Journey (Linear Path)

Brief description: Builds a coherent chain where Song A leads musically to Song B, then to Song C.

How it works:

- Requires at least 2 Seed Tracks.
- Uses Plex's sonic analysis to find the "path of least resistance" between the seeds.
- Fills the gap with tracks that bridge the sonic gap.

Inputs It Uses:

- Seed Tracks: Must have at least 2.
- Max Tracks: Acts as an approximate target length for the path.

Note: Cannot be sorted by Date/Popularity as the order is fixed by the sonic path.

7.1.11 Strict Collection

Brief description: Takes an existing collection and re-sorts/filters it without adding outside tracks.
How it works:

- Fetches all tracks from the Seed Collection.
- Applies your Playlist Parameters (filters, sort logic).
- Does *not* add any external music.

Inputs It Uses:

- Seed Collection: Mandatory.
- All Filters: (Rating, Year, Play Count) are used to prune the collection.

7.2 Seed Sources

Seed sources behave differently based on seed strategy choice.

Key Takeaway

- Playlists & Collections are just "shortcuts" to adding lots of Tracks.
- Genre Mode ignores *all* of these inputs (it only cares about the "Genre Seeds" text box).
- Strict Collection mode specifically requires a Seed Collection to work; other inputs are ignored.

7.2.1 Seed track ratingKeys (comma-separated)

To use specific tracks as inputs to generating your playlist enter one or more comma-separated Track ID (ratingKeys; e.g., 12345). To find a Track ID use the Export module and look for the “Track_ID” variable in the track level CSV file. Additionally, you can find this information within Plex by clicking the three dots on a track, selecting “Get Info”, then selecting “View XML” from the pop-up window. In the window that appears find “Track ratingKey=” and grab the specific ID number for that track.

How is this affected by Seed Strategy choice?

- Works on: Sonic Tracks Mix, Sonic Journey, Sonic Combo, Sonic Album Mix, Sonic Artist Mix, Deep Dive, and History + Seeds.
- Ignored by: Genre Mode, Strict Collection.
- Does it affect the tracklist? Yes, directly.
  - Explanation: For "Sonic Tracks" or "Journey," these specific songs are used as the targets for similarity matching. For "Album" or "Deep Dive" modes, the script looks up the *parent* album/artist of these tracks and uses those as the seeds.

7.2.2 Seed artist names (comma-separated)

To use specific artists as inputs to generating your playlist enter the artist names in a comma-separated list.

How is this affected by Seed Strategy choice?

- Works on: Sonic Artist Mix, Sonic Album Mix, Sonic Combo, Deep Dive, and History + Seeds.
- Ignored by: Genre Mode, Strict Collection, Sonic Journey, Sonic Tracks Mix.
- Does it affect the tracklist? Yes, but via expansion.
  - Explanation: In "Deep Dive," this grabs the artist's discography. In "Sonic Artist" modes, it finds other artists who sound similar to these names. Note that "Sonic Journey" ignores this input because it requires specific track waveforms to build a path, which an Artist name cannot provide.

7.2.3 Seed collection names (comma-separated)

To use specific collections as inputs to generating your playlist enter the collection names in a comma-separated list.
Note: track, album, and artist level collections will all work.

How is this affected by Seed Strategy choice?

- Works on: ALL Sonic Modes, Deep Dive, History + Seeds, and Strict Collection.
- Ignored by: Genre Mode.
- Does it affect the tracklist? Yes.
  - Explanation: Like playlists, this unpacks into a list of tracks for most modes. However, if you are using "Strict Collection" mode, this input is mandatory. That mode uses this collection as the *exclusive* source pool and ignores other inputs.

7.2.4 Seed playlist names (comma-separated)

To use specific playlists as inputs to generating your new playlist enter the playlist names in a comma-separated list.

How is this affected by Seed Strategy choice?

- Works on: Sonic Tracks Mix, Sonic Journey, Sonic Combo, Sonic Album Mix, Sonic Artist Mix, Deep Dive, and History + Seeds.
- Ignored by: Genre Mode, Strict Collection.
- Does it affect the tracklist? Yes, by unpacking into tracks.
  - Explanation: The script immediately "unpacks" the playlist into a list of individual tracks. Once unpacked, these function exactly like Seed track ratingKeys

7.3 Presets

All input parameters can be saved and loaded. Choose a Preset Name for saving and click “Save current settings as preset.” Load presets from the “Load existing preset” list and clear the form inputs back to defaults with the “Reset Inputs” button.

7.4 Custom Naming

To have your playlist appear in Plex with a custom name add a label to the “Custom playlist title (optional) box.

7.5 Playlist parameters

7.5.1 Exclude played (days)

Prevents track fatigue by excluding songs you have listened to recently.

- Input: Number of days (integer).
- Logic: Checks the viewedAt (Last Played) timestamp of every candidate track. If the track was played within the last X days, it is removed from the pool, even if it is a perfect sonic match.
- 0 = Disabled (Allows tracks played today).

7.5.2 History lookback (days)

Determines the time window used when harvesting tracks from your listening history.

- Input: Number of days (integer).
- Usage:
  - History + Seeds Mode: Defines the window for "Top Tracks."
  - Historical Ratio: If you blend history into a genre playlist, this setting defines how far back the script looks to find your "favorites."
  - Default: 30 days.

7.5.3 Max tracks

The target length for the final playlist.

- Input: Integer (e.g., 50, 100).
- Note: In "Sonic Journey" mode, this acts as an approximate target, but the actual path length may vary slightly depending on how the algorithm connects the start and end tracks.

7.5.4 Sonically similar per seed

Controls the "width" of the net cast when searching for similar music in Sonic modes.

- Input: Integer (Standard default is 20).
- Logic: For every seed track you provide, the script asks Plex for the top *N* sonically similar tracks.
- Impact:
  - Lower values (e.g., 5-10): Tighter matches, very similar to the seed, but less variety.
  - Higher values (e.g., 100): More variety and discovery, but potential for less relevant matches.

7.5.5 Deep Dive Target (Seeds per Artist)

Specific to Deep Dive mode. Controls how aggressively the script mines an artist's discography to find albums.

- Input: Integer (Tracks per artist).
- Logic: When you seed an artist, the script grabs this many "top tracks" from them to identify which albums to pull from.
- Impact: A higher number ensures more albums from that artist are covered (e.g., finding the "Deep Cuts" album, not just the "Greatest Hits" album).

7.5.6 Historical ratio (fraction of tracks from history)

Determines the balance between "New Discovery" (Seeds/Genre) and "Nostalgia" (Your History).

- Input: Decimal between 0.0 and 1.0.
- 0.0 (Pure Discovery): The playlist is built entirely from your seeds or genre rules.
- 0.3 (30% History): The script reserves 30% of the playlist slots for tracks from your listening history (filtered by the History Lookback settings).
- 1.0 (Pure History): The playlist is strictly your top tracks, essentially ignoring other seeds.

7.5.7 Explore vs. exploit (popularity)

Determines if you get "hits" or "deep cuts" from that genre.

Range:

- 1.0 (Exploit): Strictly picks the highest-scoring tracks (the "Best" matches).
- 0.0 (Explore): Completely shuffles the candidate pool (Pure Random).

Recommendation: A value of 0.7 ensures high quality tracks while allowing for some surprises. If you want less “hits” and more “deep cuts” use a value of 0.3 or lower.

7.5.8 Sonic Smoothing (Gradient Sort)

A post-processing step that reorders the final playlist to ensure optimal flow.

- Checkbox: On/Off.
- Logic: It does not select *which* songs are in the playlist, only *where* they sit. It uses a "Scout + Chain" method to arrange tracks so that each song is sonically similar to the one before it, creating a seamless listening experience rather than a jarring shuffle.

7.5.9 When explicit seeds are too few, fill from (history/genre)

The fallback safety net. If your specific inputs (e.g., "Sonic matches for this obscure track") result in fewer songs than your Max Tracks target, this setting determines how to fill the empty space.

- Options:
  - History: Fills the remainder with your top played tracks (subject to History filters).
  - Genre: Fills the remainder with random tracks matching the Genre Seeds (if provided).
  - None: The playlist will simply be shorter than requested.

7.6 Rating filters

Track ratings are measured on 10 pt scale, where 4.5 stars is a rating of 9 and 5 stars is a rating of 10.

7.6.1 Min track rating

Set minimum track rating for inclusion on a playlist (0-10).

7.6.2 Min album rating

Set album track rating for inclusion on a playlist (0-10).

7.7.3 Min artist rating

Set minimum artist rating for inclusion on a playlist (0-10).

7.7.4 Allow unrated items (Track/Album/Artist)

If “Allow unrated items” is checked then items (tracks/albums/artists) can still appear on a playlist.

If Min Track Rating > 0:
- It checks the Track's rating.
- If the rating is None (unrated) and "Allow Unrated" is OFF, the track is rejected.

If Min Album Rating > 0:
- It looks up the Album.
- If the Album's rating is None and "Allow Unrated" is OFF, the track is rejected (even if the track itself is rated 5 stars).

If Min Artist Rating > 0:
- It looks up the Artist.
- If the Artist's rating is None and "Allow Unrated" is OFF, the track is rejected (even if the track and its album are rated 5 stars).

Tip: If you want to omit low rated tracks but include unrated tracks, set a minimum rating score and check the allow unrated items box.

7.7 Play count filters

Based on number of plays in Plex (viewCount). Use –1 to ignore a bound. Example: min=0, max=0 will include only never-played tracks.

Variables:
- Min play count (–1 = no minimum)
  - Set minimum number of plays for each track.
- Max play count (–1 = no maximum)
  - Set maximum number of plays for each track.

7.8 Year & duration filters (album-based year)

Inputs of 0 represents no minimum for each setting.

Variables for setting album ranges:
- Min album year (0 = none)

    - e.g., “1965” (or later)
- Max album year (0 = none)
    - e.g., “2000 (or earlier)

Variables for setting track duration ranges:

- Min track duration (sec, 0 = none)
- Max track duration (sec, 0 = none)

7.9 Date Added Bias

Determines if you get more newly added tracks or older tracks (those in your library longer)

Range:

- 1.0 = Prefer the newest tracks available in the current selection of candidate tracks.
- 0.0 = Ignore date when selecting tracks for playlist.

7.10 Artist / album caps

Set a cap for how many tracks from specific albums or artists are allowed (0 = no cap).

Variables:

- Max tracks per artist (0 = no cap)
- Max tracks per album (0 = no cap)

7.11 History filters (for history-based seeds & fallbacks)

When do "History Filters" apply?
They apply only to tracks pulled from your listening history in these three specific scenarios:

- Historical Ratio: If you are running a "Genre" playlist but have Historical Ratio set to 0.3 (30%), the script pulls 30% of the tracks from your listening history. Those specific tracks will be filtered by History Min Rating / History Max Play Count.
- History Mode: If your Seed Mode is set to "History + Seeds", the history portion obeys these filters.
- Fallback: If your Genre seeds generate 0 results and the script falls back to History, the fallback tracks obey these filters.

Tracks harvested via **Genre Seeds** (or Sonic similarity, or Deep Dive) ignore the "History Filters."

Variables:

- History: min track rating (0–10)
- History: max play count (–1 = no max)

7.12 Genre & collections filters

These filters allow you to include or omit tracks based on genre or collection inclusion.
Note: These inputs behave differently based on seed strategy choices.

7.12.1 Genre seeds (comma-separated, track or album)

List genre tagging for tracks and albums that you want included in your playlist (note: artist level genre tagging is ignored for this input).

How is this affected by Seed Strategy choice?

- Works on: Genre Mode (and "Auto" if it guesses Genre).
- Ignored by: Deep Dive, Sonic Modes, History, Strict Collection.
- Does it affect the tracklist? Only if you are in Genre Mode.
    - *Explanation:* If you are in "Sonic Journey" mode and you type "Rock" here, the script ignores it. It does *not* force your Sonic Journey to be Rock songs. It is strictly used to *start* the playlist in Genre Mode.

7.12.2 Include only collections (comma-separated)

List any collections in your plex library that a track must be included within to be present in your generated playlist.

How is this affected by Seed Strategy choice?

- Works on: ALL Modes.
- Does it affect the tracklist? Yes, drastically.
    - *Explanation:* This acts as a strict gatekeeper. If you run a "Sonic Journey" or "Deep Dive" but put "My Favorites" in this box, the script will generate the playlist normally, but then delete every song that is *not* in your "My Favorites" collection.

7.12.3 Exclude collections (comma-separated)

List any collections in your plex library that will omit any track that is included within them (e.g., “Christmas” or “Holiday” if you don’t want music from this collection appearing).

How is this affected by Seed Strategy choice?

- Works on: ALL Modes.
- Does it affect the tracklist? Yes.
    - *Explanation:* This applies the final "cleanup" to every mode. If you exclude the "Christmas" collection, those songs are removed regardless of whether they came from a Sonic match, a Deep Dive, or your History.

7.12.4 Exclude genres (comma-separated)

List any genres you do not want present in your playlist.
Note: This variable checks all levels for each track (track, album, and artist metadata).

How is this affected by Seed Strategy choice?

- Works on: ALL Modes.
- Does it affect the tracklist? Yes.
    - *Explanation:* This runs a "scorched earth" check on Track/Album/Artist tags. It will strip these genres out of any playlist type (Sonic, History, etc.).

7.12.5 Genre strict (enforce genres against genre seeds)

Choose whether or not to be strict with genre inclusion. If checked, only genres listed in the genre seeds box will be allowed. If unchecked, then the percentage of tracks outside the listed genres will map onto inputted slider value.

Hierarchy of checks when "Genre Strict" is on:

1. Track Genres (First Priority):
    - If the specific song is tagged "Rock", it passes, even if the Album is "Pop".
2. Album Genres (Second Priority):
    - If the song has no tags, it checks the Album. If the Album is "Rock", it passes.
3. Artist Genres (Third Priority):
    - If the Song and Album are empty, it checks the Artist. If the Artist is tagged "Rock", it passes.

How is this affected by Seed Strategy choice?

- Works on: ALL Modes, assuming you provided at least one genre tag in the Genre seeds box.

8. Automation & Scheduling (Unraid)

You can automate the generation of your playlists so they refresh with new music automatically (e.g., waking up to a fresh "Morning Coffee" mix every day).
For Unraid users, the recommended method is using the **User Scripts** plugin.

8.1 Setting up a User Script

1. **Prerequisite:** Install the **User Scripts** plugin via the Unraid Apps tab.
2. **Create a Preset:** Ensure you have saved your desired settings as a **Preset** inside the Music-Manager-for-Plex web interface (e.g., "Indie Mix").
3. **Add New Script:**
    - Go to **Settings > User Scripts**.
    - Click **"Add New Script"**.
    - Name it (e.g., Playlist Creator - Indie Mix).
4. **Enter the Script:** Click the **gear icon** next to your new script and select **"Edit Script"**. Paste the following code:

```
#!/bin/bash
# Run the Playlist Creator inside the Docker container using a saved preset
# Replace "Indie Mix" with your exact Preset Name (keep the quotes)

docker exec Music-Manager-for-Plex python /app/Extras/Scripts/playlist_creator.py --preset "Indie Mix"

# or to schedule regular metadata exports
docker exec Music-Manager-for-Plex python /app/Extras/Scripts/xport_library_metadata.py
```

5. **Save Changes.**

8.2 Scheduling (Cron Syntax)

To run the script automatically, you need to define a schedule using **Cron Syntax** in the "Custom" schedule box.

A Cron schedule consists of 5 fields separated by spaces:
Minute
Hour
Day_of_Month
Month
Day_of_Week

Breakdown:

- **Minute:** 0-59
- **Hour:** 0-23 (24-hour format, e.g., 5 = 5 AM, 17 = 5 PM)
- **Day of Month:** 1-31
- **Month:** 1-12
- **Day of Week:** 0-7 (0 and 7 are Sunday, 1 is Monday, etc.)

Common Examples:

Schedule	Syntax	Description
Mon, Wed, Fri at 5:30 AM	30 5 * * 1,3,5	Runs at minute 30, hour 5, on days 1, 3, and 5.
Every Day at 5:00 AM	0 5 * * *	Runs daily at 5 AM.
Every Sunday at Midnight	0 0 * * 0	Runs once a week on Sunday.
Every 3 Hours	0 */3 * * *	Runs at minute 0 past every 3rd hour.

Once your schedule is set, click **Apply**. Your playlist will now regenerate in the background according to your rules.

9. Safety & Troubleshooting

- Token Security: Never commit your config.txt to GitHub.
- "Sonic Analysis" Error: If the Playlist Creator fails, ensure you have enabled "Sonic Analysis" in your Plex Library settings and that the scheduled task has completed processing your music.
- Bulk Edits: Writes to Plex are potentially destructive. Always Export a backup CSV before running a bulk update script.