

MediaTools Video Downloader

Comprehensive User Guide

Version: 1.0.0

Developer: Bala

Organization: MediaTools

Email: bala.lv.555@gmail.com

GitHub: <https://github.com/MediaTools-tech/mediatools>

Website: <https://mediatools.tech>

Table of Contents

1. [Introduction](#)
 2. [Installation](#)
 3. [Getting Started](#)
 4. [Downloading Videos](#)
 5. [Settings and Features](#)
 6. [FAQ and Troubleshooting](#)
 7. [Contributing and Feedback](#)
 8. [Privacy and Legal Information](#)
 9. [Contact and Support](#)
 10. [Version History](#)
-

1. Introduction

MediaTools Video Downloader is a powerful, cross-platform desktop application designed to download videos from popular platforms. Built with user-friendliness in mind, it provides extensive customization options while maintaining simplicity for everyday use.

Key Features

- **Auto-update option** – Auto update to latest yt-dlp.
- **Multi-platform support** - Download yt-dlp supported 1000+ sites
- **Format flexibility** - Choose between MKV (highest quality) or MP4 (best compatibility)
- **Smart format fallback** - Automatically retries downloads with alternative formats if the preferred one fails
- **Quality control** - Select optimal quality settings for your needs
- **Speed management** - Control download speeds to manage bandwidth
- **Queue system** - Download multiple videos sequentially
- **Thumbnail support** - Embed video thumbnails for easy identification
- **Cookie integration** - Access private or membership-only content
- **Cross-platform** - Works on Windows, Linux, and macOS

Technology Stack

This application leverages two industry-standard open-source tools:

- **yt-dlp** (<https://github.com/yt-dlp/yt-dlp>) - Advanced video downloading engine
- **FFmpeg** (<https://ffmpeg.org>) - Media processing and format conversion

Both dependencies are automatically managed by MediaTools Video Downloader, before download prompts user for permission to download.

2. Installation

2.1 System Requirements

Minimum Requirements

Windows:

- Windows 10 or later (64-bit)
- 4GB RAM
- 500MB free disk space (plus space for downloaded videos)
- Internet connection

Linux:

- Ubuntu 20.04+ or equivalent distribution
- 4GB RAM
- 500MB free disk space
- Internet connection
- Python 3.8+ (usually pre-installed)

macOS:

- macOS 10.14 (Mojave) or later
- 4GB RAM
- 500MB free disk space
- Internet connection

Recommended Requirements

- 8GB RAM or more
- SSD for faster video processing
- Broadband internet connection (10 MBps or faster)

2.2 Installation Steps

Step 1: Choose Your Version & Download

1. Visit the MediaTools GitHub releases page and choose between two versions for each platform:
 - Single Executable:
 - a. Clean appearance
 - b. Application data stored in User/AppData/ (Windows) or ~/.config/ (Linux)
 - Portable Version:
 - a. One portable dir – Self contained
 - b. All application files reside in the _internal/ folder alongside the executable. Can be placed anywhere (USB drives, external drive, etc.)

Download the appropriate installer for your operating system:

- Windows: mt-vdl.exe
- Linux: mt-vdl (executable)
- macOS: mt-vdl.app or mt-vdl (executable)

Step 2: Desktop Shortcut (Optional)

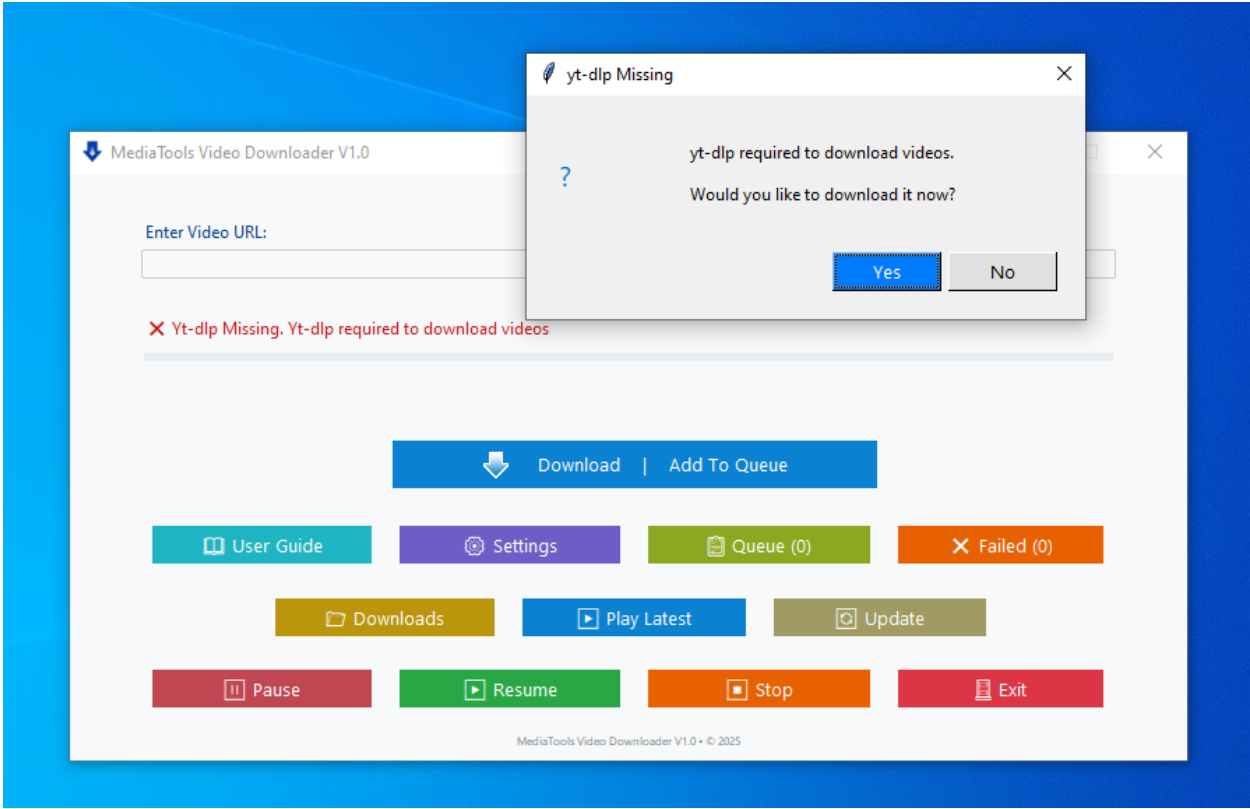
The application will offer to create a desktop shortcut for easy access. You can:

- Accept to create the shortcut now
- Decline and create it later

Step 3: Install Dependencies

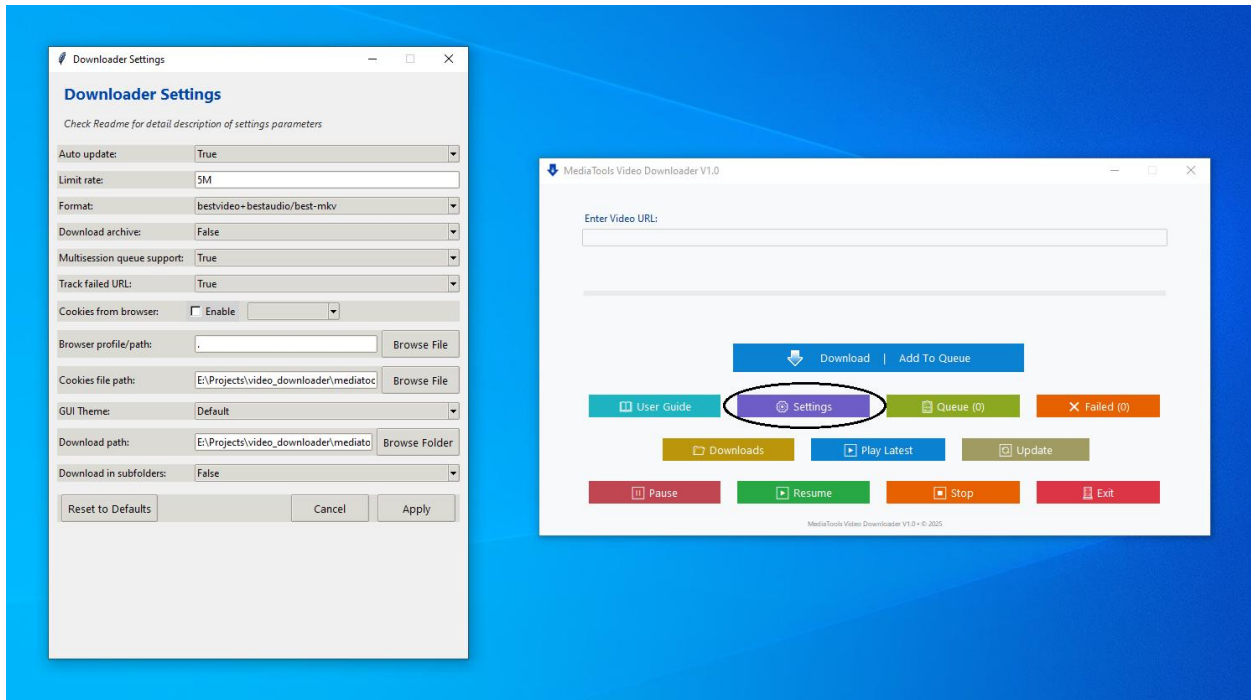
On first launch, MediaTools Video Downloader will:

1. Check for yt-dlp and FFmpeg
2. Prompts you for permission to download, if not present
3. Configure them for optimal performance



Step 4: Configure Initial Settings

- Reset to Defaults as initial settings
- Restart app



Step 5: Configure Preferred Settings

- After initial default settings and restarting app, if required, you may set your preferred settings like Download location, Default video format, Download speed limit etc

2.3 Platform-Specific Notes

Windows

- May require administrator privileges for first run
- Windows Defender/ Antivirus might flag unknown executables (false positive)
- Add exception if needed: Settings → Security → Windows Security
- Desktop shortcut creation may require permissions

Linux

- Ensure execute permissions: `chmod +x mt-vdl`
- Some distros require additional dependencies
- Desktop file may need to be marked as trusted
- Check file manager thumbnail settings for video previews

macOS

- May show "unidentified developer" warning
 - Allow in System Preferences → Security & Privacy
 - Grant permissions for network access
 - Some features require full disk access permission
-

3. Getting Started

3.1 Launching the Application

Windows:

- Double-click the desktop shortcut, or
- Run `mt-vdl.exe` from the installation folder

Linux:

- Click the desktop icon, or
- Run `./mt-vdl` from terminal, or
- Launch from applications menu

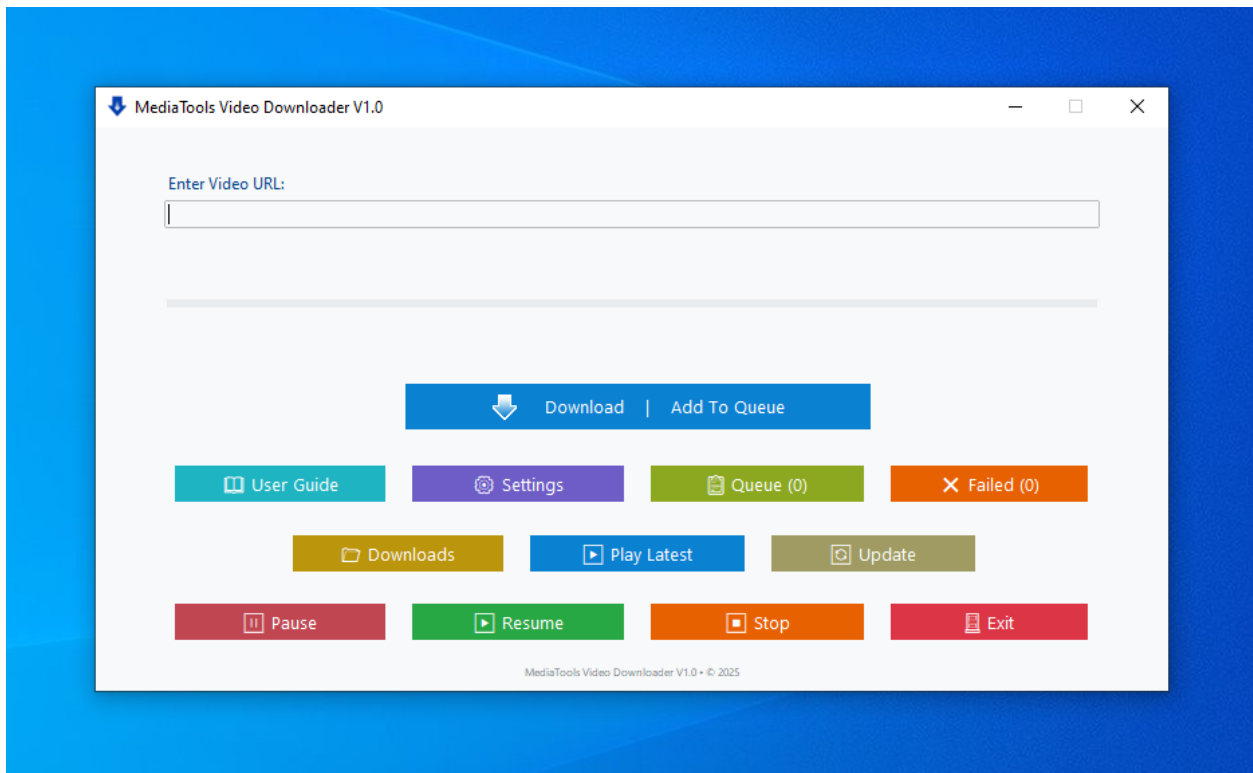
macOS:

- Double-click the application icon, or
- Open from Applications folder

3.2 Interface Overview

Window Layout

The MediaTools Video Downloader interface is organized into several key areas:



1. URL Input Section

URL Input Field: Primary field for pasting video URLs

URL Management

- Single URL Processing: Paste and download immediately
- Queue System: Multiple URLs processed sequentially
- Clipboard Integration: Right-click paste option

2. Progress Visualization

Progress Bar: Real-time status display of overall download progress of active download.

3. Download Control Buttons

- Download | Add To Queue: Process current URL or add to download queue
- Docs: Open documentation folder with all help files
- Settings: Access application configuration and themes
- Queue: View and manage download queue
- Failed URL: Review and retry failed downloads
- Downloads: Open file explorer to download location
- Play Latest: Launch the most recently downloaded video
- Update: Check for yt-dlp/FFmpeg updates
- Pause: Temporarily halt all active downloads (preserves progress)
- Resume: Continue paused downloads from interruption point
- Stop: Cancel all current operations and clear active downloads
- Exit: Safely close the application

Document Access

Docs button: Opens the local `docs/` folder containing:

- README.md (Quick start guide)
- LICENSE (Application license)
- UserGuide.pdf (Complete documentation)
- FAQ.md (Troubleshooting guide)

Download Controls

- Pause/Resume: Maintains download state for resumable formats
- Stop: Complete cancellation with cleanup
- Real-time Progress: Continuous feedback during operations

Quick Access Features

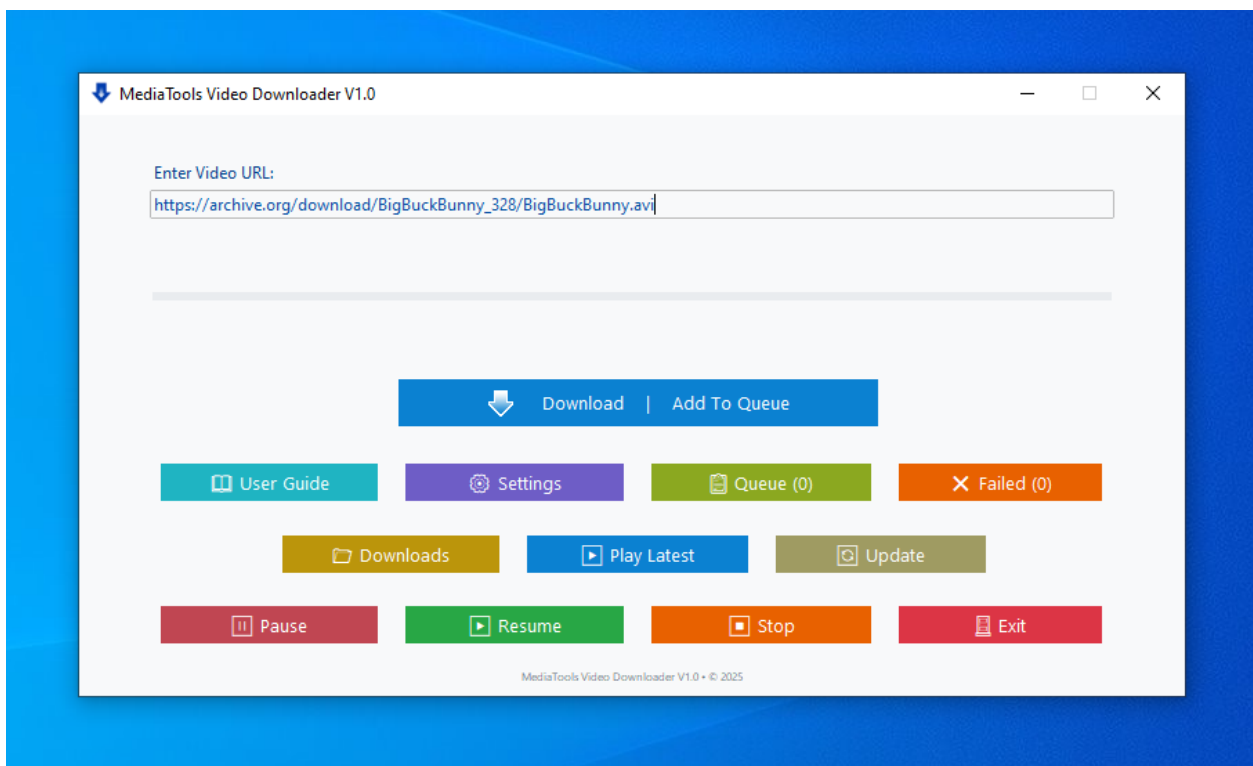
- Open Download Folder: Direct access to downloaded files
- Play Latest Video: Quick preview of most recent download
- Update Tools: Ensure yt-dlp/FFmpeg are current

Theme & Customization

- Accessible via the Settings button:
- Theme Selection: Choose between Dark, Light, and Minimalist themes
- Instant Application: Theme changes apply immediately
- Persistent Settings: Preferences saved between sessions

3.3 First Download

1. Launch MediaTools Video Downloader
2. Copy a video URL from any yt-dlp supported platform.
3. Paste the URL in the input field at the top
4. Click the "Download" button
5. Watch the progress in the status area
6. Click downloads button on App to check your downloads folder



4. Downloading Videos

4.1 Basic Download Process

Single Video:

1. Copy the video URL from your browser
2. Paste it into the URL input field
3. Click "Download"
4. Monitor progress in the status bar

Playlist Download:

1. Paste the playlist URL
2. The application automatically detects it's a playlist
3. Click "Download"
4. All videos in the playlist will be downloaded sequentially
5. Progress bar shows playlist real time download status

Supported Platforms

yt-dlp supports over 1000 websites. Check yt-dlp official link for detail list
<https://github.com/yt-dlp/yt-dlp/blob/master/supportedsites.md>

Download Progress Information

During download, you'll see:

- **Current File:** Which file is downloading (Url) and **Percentage Complete:** 0% to 100%

Pause and Resume

Note: Pause and Resume functionality depends on the video platform's support for partial downloads and not supported for Format setting - "bestvideo+bestaudio/best-mp4".

Pausing a Download:

1. Click the "Pause" button during download
2. The download stops and progress is saved, partial files (.part) are retained.

Resuming a Download:

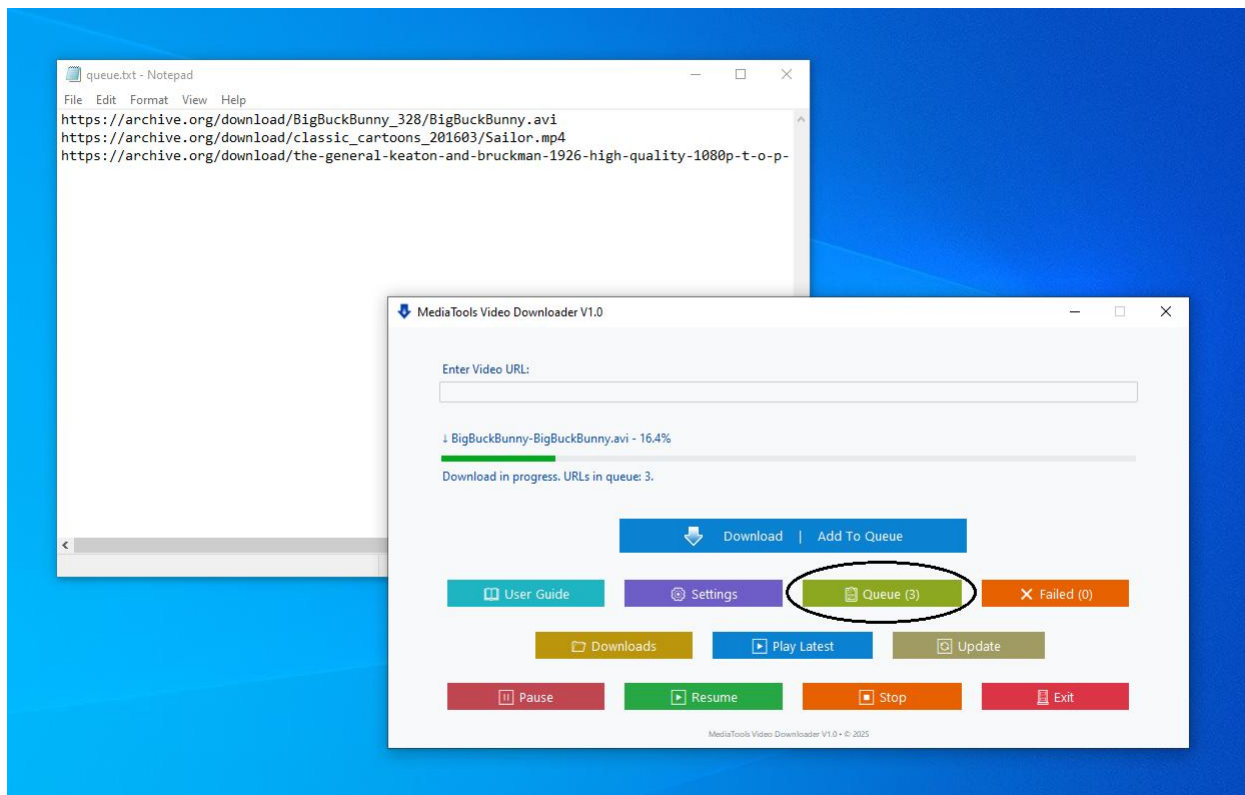
1. Click "Resume" button
2. Download continues from where it stopped

4.2 Queue Management

Understanding the Queue

The queue system allows you to:

- Download multiple videos sequentially
- Manage download order
- Pause/resume entire queues*
- Track failed downloads*



Adding to Queue

Manual Addition:

1. There are two ways to create queue
 1. Paste URL in url entry field and click "Download", every time you add new URL it will be added to queue.
 2. You can also directly paste multiple URLs in Queue.txt (even works with comma separated URLs) and all these URLs will be added to queue. After adding URLs in Queue.txt, click "Download". You can directly access Queue.txt by clicking "Queue" button on App.
2. If a download is in progress, the new URL is queued
3. Downloads process in order (FIFO - First In, First Out)

Playlist Handling:

- Playlists are automatically identified and all videos in playlist downloaded sequentially.
- You can see pending video count from playlist in the download status.

Queue Operations

View Queue:

- The Queue button on App opens Queue.txt file with list of URLs in queue.
- Completed downloads are automatically removed from Queue.txt

Manage Queue Items:

- Pause or Stop (Terminate) download ongoing download.
- Stop download will remove URL from queue
- Edit Queue.txt file to modify queue.

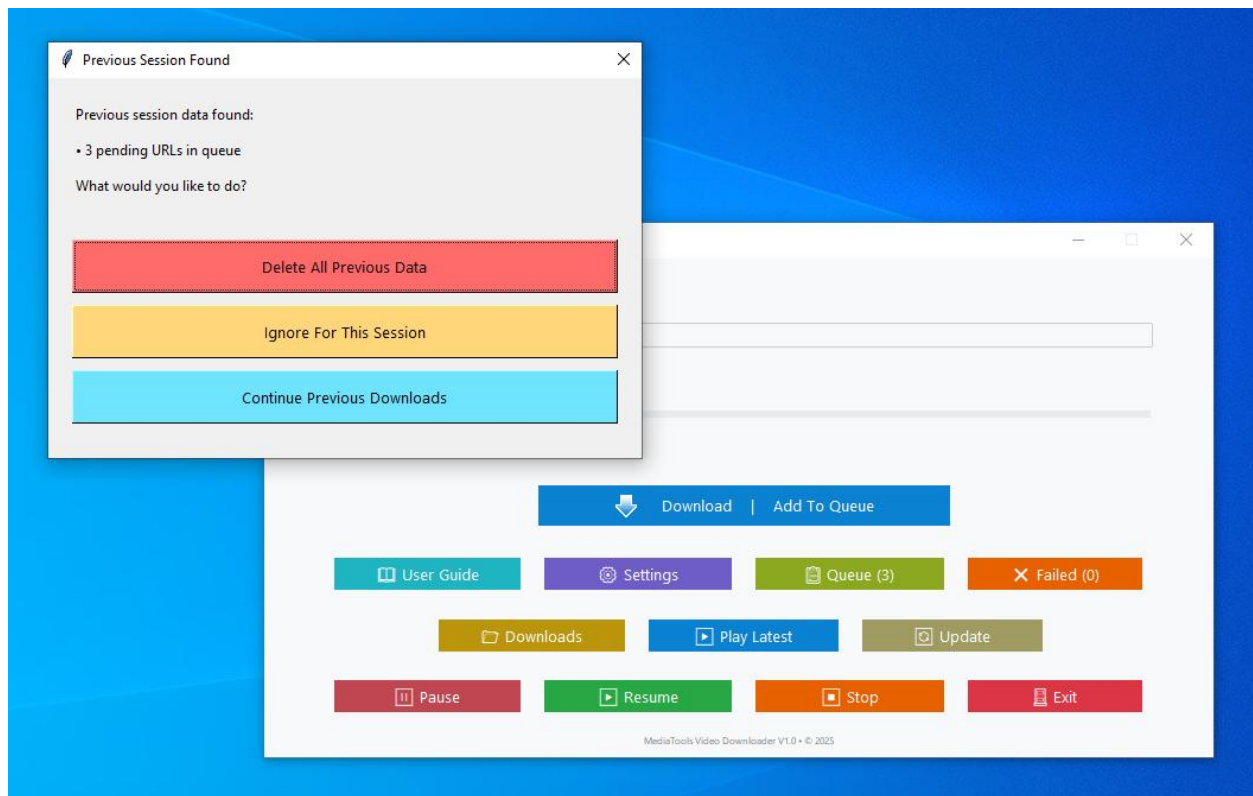
Queue Persistence – with Setting option “Multisession queue support” Enabled :

- Can be enabled with Settings option: “Multisession queue support”.
- Failed downloads are tracked separately (if enabled).

Multi-Session Queue Support

When enabled in settings, this feature allows:

- At startup provides option to continue previous download session.
- Continue downloading queue from previous session.
- If enabled and supported, partially downloaded last video from previous session will be resumed.



5. Settings And Features

5.1 Settings Configuration

Access settings through: **Settings Button on App**

General Settings

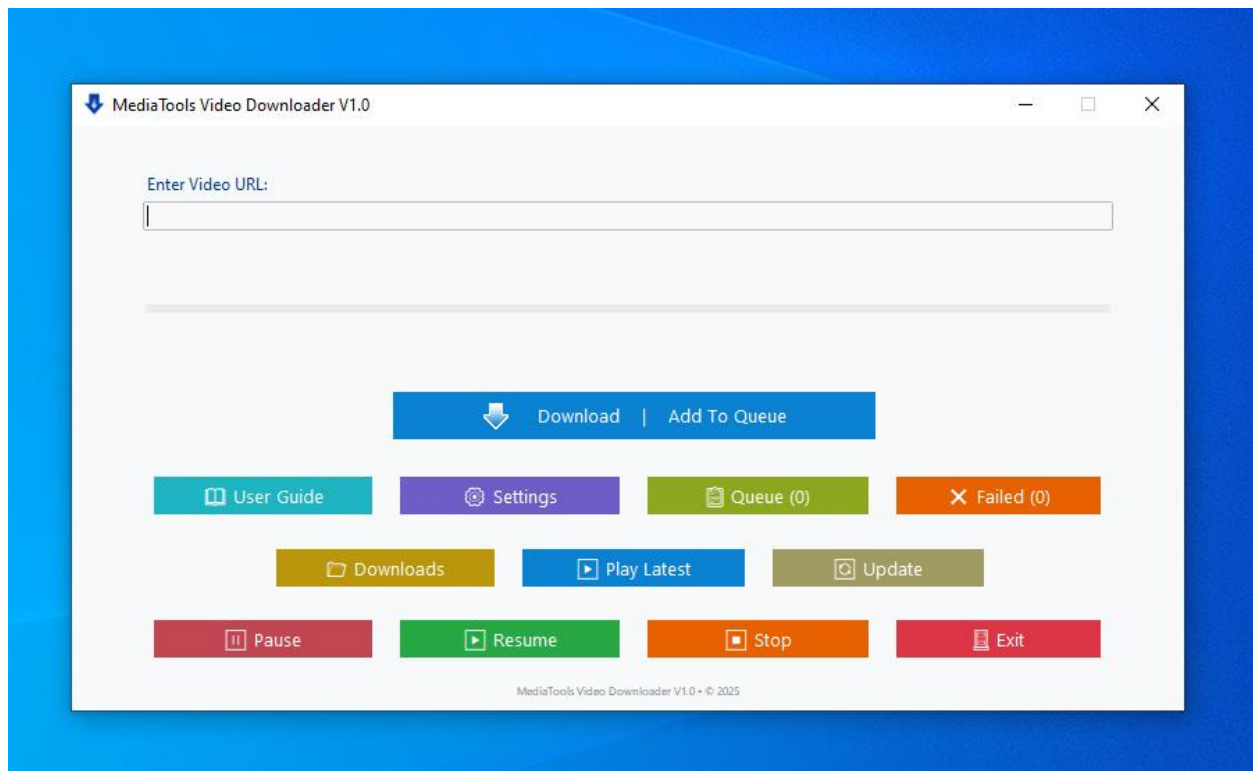
Auto Update (yt-dlp)

- **Default:** Enabled (Highly recommended)
- **Description:** Automatically updates yt-dlp to the latest version on startup
- **Recommendation:** Keep enabled to ensure compatibility with video platforms

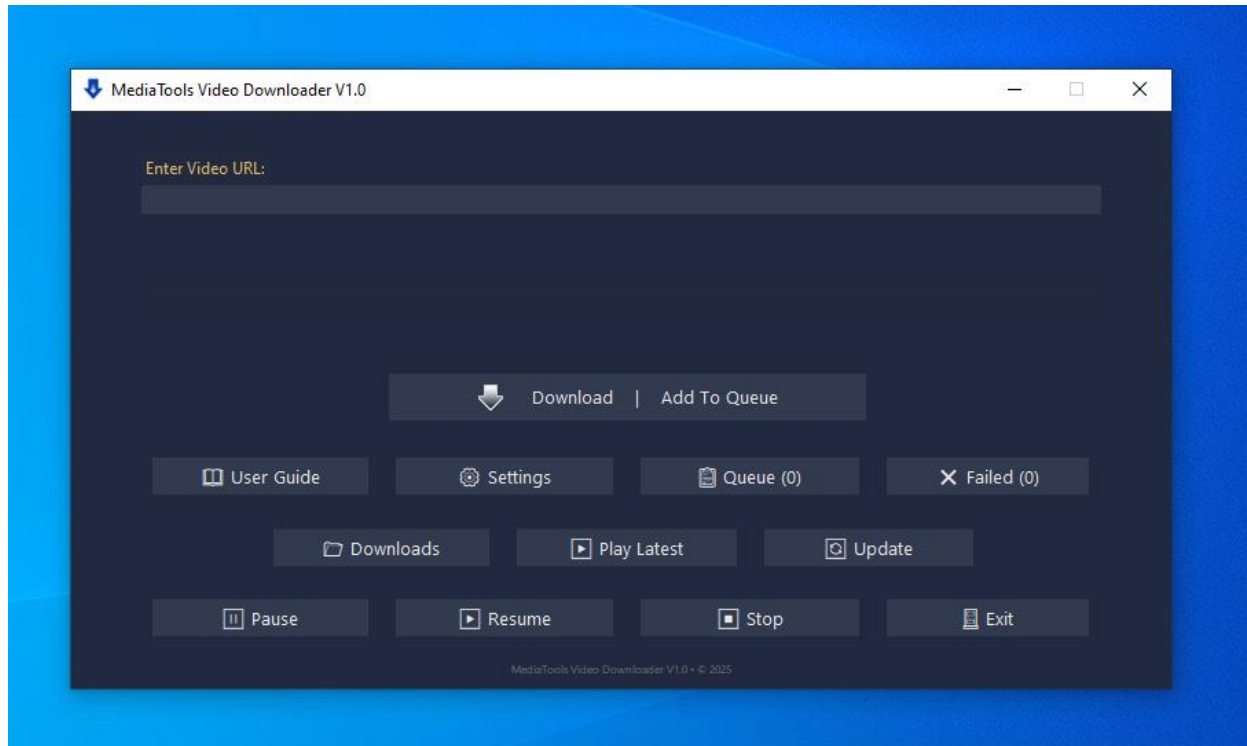
GUI Theme

- **Options:** Default, Dark, Unicolor, Minimalist
- **Description:** Changes the application's visual appearance
- **Note:** Theme selection persists across sessions

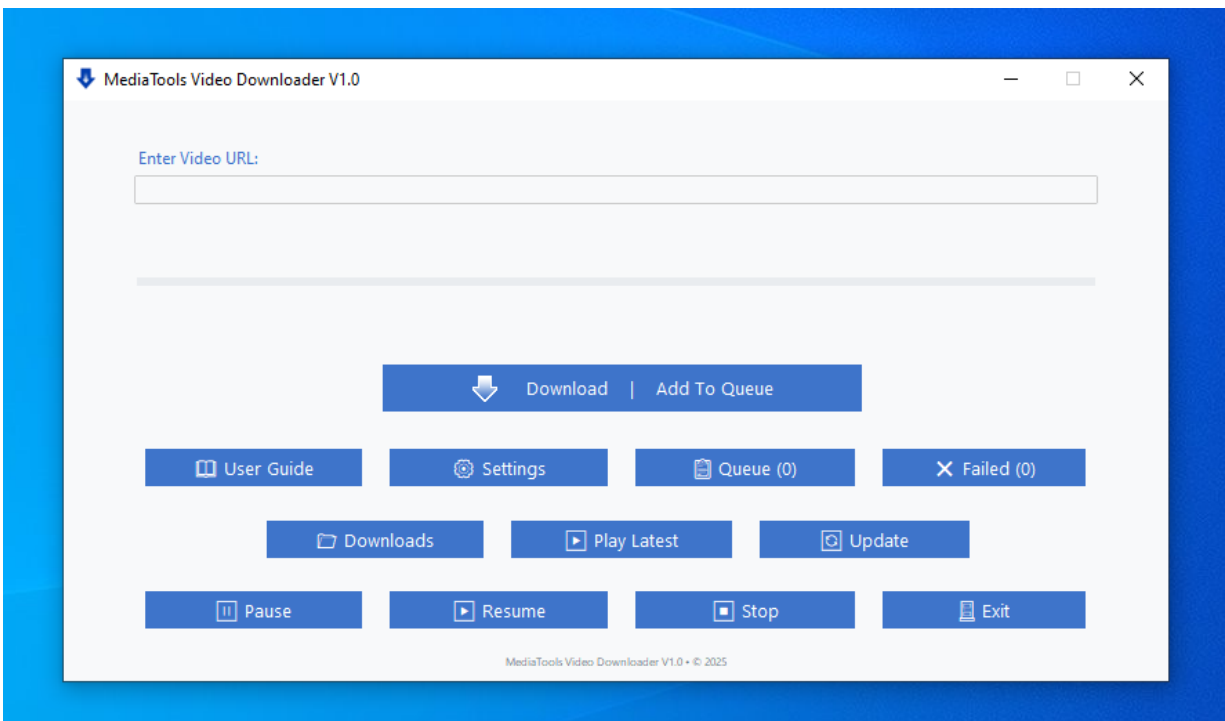
Theme: Default



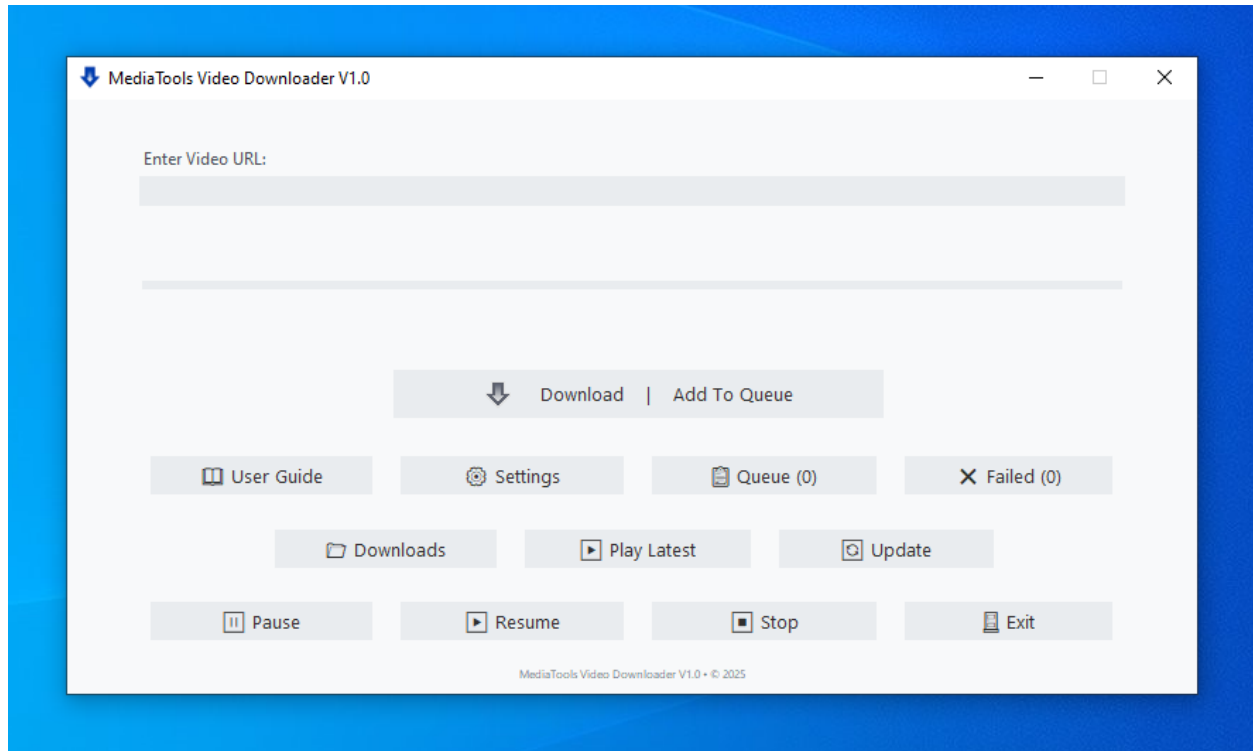
Theme: Dark



Theme: Unicolor_1



Theme: Minimalist_1



Download Settings

Download Speed Limit

- **Default:** 5M (5 megabytes per second)
- **Description:** Limits download speed to prevent network saturation
- **Usage:**
 - Set lower (1M-2M) for shared networks
 - Set higher (10M+) some platforms/sites may block or restrict high speed downloads.

Downloads Directory

- **Default:** [User]/Videos/ or application folder
- **Description:** Where downloaded videos are saved
- **Options:**
 - Click "Browse" to change location
 - Create new folders as needed
 - Ensure sufficient disk space

Enable Download Archive

- **Default:** Disabled
- **Description:** Tracks downloaded videos to avoid re-downloading
- **When Enabled:**
 - Creates download_archive.txt in data folder
 - Skips previously downloaded videos automatically

Format Settings:

Stream and Merge Format

- **Options:**
 - bestvideo+bestaudio/best-mkv - Best quality in MKV container
 - bestvideo+bestaudio/best-mp4 - Best quality in MP4 container
 - b - Single best quality format

Understanding Format Options:

MKV Format:

- Pros: Highest quality, supports all codecs, better for archiving
- Cons: Larger file sizes, thumbnail support varies
- Best for: Quality-focused users, video editing, archival

MP4 Format:

- Pros: Universal compatibility, smaller files, reliable thumbnails
- Cons: Slightly lower quality ceiling, limited codec support
- Best for: Mobile viewing, sharing, streaming devices

Single Best (b):

- Pros: Fastest download, single file, reliable
- Cons: Quality depends on platform's pre-encoded formats
- Best for: Quick downloads, low storage space

Format	Description	Best For
MP4	Merges best video and audio streams	Max compatibility(phone, social media)
MKV	Merges best video and audio streams	<u>Highest quality</u> preservation, pause/resume support
b (best)	Single pre-merged file when available	Fastest downloads, no merging required

Queue Settings

Multisession Queue Download Support

- **Default:** Enabled
- **Description:** Saves queue state between application sessions
- **When Enabled:**
 - Queue persists after closing app
 - Resumes unfinished downloads
 - Maintains download order

Track Failed URLs

- **Default:** Enabled
- **Description:** Saves URLs that failed to download
- **Benefits:**
 - Review failed downloads later
 - Retry failed URLs easily
 - Identify problematic sources

Advanced Settings

Platform-Specific Download Folders

- **Default:** Disabled
- **Description:** If enabled, creates separate sub-folders for each platform (platform names to be specified in “**Subfolder Domains**” for platforms/domains to group videos)
- **When Enabled:**
 - YouTube videos → downloads/youtube/
 - Twitter videos → downloads/twitter/
 - Other platforms → downloads/other/

Subfolder Domains

- **Default:** Empty
- **Description:** Custom domain-to-folder mappings
- **Format:** domain1.com:folder1,domain2.com:folder2
- **Example:** youtube.com:YT,vimeo.com:Vimeo

Cookie Settings

Cookies allow downloading private or membership-only content.

Cookie File

- **Description:** Import cookies from a text file
- **Format:** Netscape cookie format
- **How to Get:**
 1. Use browser extension (e.g., "Get cookies.txt")
 2. Export cookies from target website
 3. Save as cookies.txt
 4. Browse and select in settings

Cookies from Browser

- **Description:** Extract cookies directly from installed browser
- **Supported Browsers:**
 - Brave
 - Chrome
 - Chromium
 - Edge
 - Firefox
 - Opera
 - Safari
 - Vivaldi
 - Whale

Browser Selection:

- Choose your browser from dropdown
- Optionally specify browser profile
- Cookies are read automatically during download

Browser Profile:

- **Default:** Default profile used
- **Custom:** Specify profile name for multi-profile browsers
- **Example:** "Profile 1", "Work", "Personal"

Important Notes:

- Cookies may expire; refresh periodically
- Close browser before extracting cookies
- Some platforms detect and block cookie usage

5.2 Performance Tips

Optimizing Download Speed

1. **Adjust Speed Limit:** Know what download speed will work best for your requirements.

Managing Disk Space

1. **Choose Appropriate Format:** MP4 is smaller than MKV
2. **Monitor Available Space:** Ensure sufficient room before large downloads
3. **Regular Cleanup:** Delete paused and skipped video files (.part, webp etc.)
4. **Platform-Specific Folders:** Organize by platform for easier management

Queue Management Best Practices

1. **Prioritize Important Downloads:** Place urgent items at queue front
2. **Use Download Archive:** Avoid re-downloading same content

5.3 Updates and Maintenance

Keeping yt-dlp Updated: Setting Auto-Update in settings is highly recommended

Auto-Update:

- Enable in Settings → Auto Update
- Checks for updates on startup
- Downloads automatically if available

Why Update:

- yt-dlp frequently releases updated versions as video platforms frequently change their APIs
- New platform support added
- Bug fixes and performance improvements
- If Auto-Update not enable, yt-dlp can be updated manually through Update button on App

Application Updates

Checking for Updates:

1. Visit GitHub releases page: <https://github.com/MediaTools-tech/mediatools>
2. Download latest version if available
3. Install over existing version (settings preserved)

5.4 Advanced Features

Thumbnail Embedding

What It Does:

- Embeds video thumbnail inside the video file
- Allows preview in file managers and media players
- Useful for organizing large video libraries

Format Compatibility:

- **MP4:** Reliable thumbnail support
- **MKV:** Variable support depending on player

Settings:

- Enabled by default for MP4 format
- Automatically converted to compatible format
- Thumbnails stored with video metadata

Download Naming

Video files are automatically named using this pattern:

[Title]-[Video_ID].[Extension]

Example:

How_to_Use_MediaTools-dQw4w9WgXcQ.mp4

Features:

- Special characters removed for compatibility
- Title truncated to 100 characters
- Video ID ensures uniqueness
- Prevents filename conflicts

Restricted Filenames

Automatically Enabled:

- Removes special characters: < > : " / \ | ? *
- Replaces spaces with underscores
- Ensures cross-platform compatibility
- Prevents filesystem errors

Download Archive

It's a text file containing IDs of downloaded videos, not names.

Purpose: Prevents re-downloading videos you already have.

How It Works:

1. After successful download, video ID is saved
2. On next download attempt, ID is checked
3. If found, download is skipped
4. Saves time and bandwidth

Archive Location:

[App Data]/data/download_archive.txt

Use Cases:

- Downloading updated playlists
- Channel subscriptions
- Avoiding duplicates

Failed URL Tracking

When Enabled:

- Failed downloads are logged with reason
- URLs saved to failed_url.txt
- Allows bulk retry later
- Helps identify problematic sources

Accessing Failed URLs:

- Current session Failed_url.txt can be accessed by clicking “Failed” button on App or navigate to application data folder for Failed_url.txt and Failed_url_old.txt (for failed URLs from previous session).



6. FAQ and Troubleshooting

6.1 FAQ

General Questions

Q. Is MediaTools Video Downloader free? A. Yes, MediaTools Video Downloader is open-source and free to use under the MIT License.

Q. Which websites are supported? A. yt-dlp supports 1000+ sites. For the complete current list, check: <https://github.com/yt-dlp/yt-dlp/blob/master/supportedsites.md>

Q. Do I need to install Python? A. No, the application is self-contained and includes all necessary components.

Q. Is my data collected or shared? A. No. The application does not collect, store, or transmit any personal data. All processing happens locally on your computer.

Q. My antivirus flags the application - is it safe? A. Yes, this is a common false positive with applications that download executables (yt-dlp/FFmpeg). The application is open-source and completely safe. You may need to add an exception in your antivirus software.

Q. Why can't I download from some websites? A.

- The website may not be supported by yt-dlp, check yt-dlp supported sites [list](#).
- Regional restrictions may block access
- Some sites require cookies for authentication

Installation & Setup

Q. What if I skip downloading yt-dlp and ffmpeg? A.

yt-dlp: The application cannot function without yt-dlp. You have two options:

- Either allow the application to automatically download yt-dlp when prompted
- Or manually download yt-dlp from [yt-dlp official repository](<https://github.com/yt-dlp/yt-dlp>) and place the executable in the `bin/` folder

FFmpeg: The application can work without FFmpeg, but with limitations:

- Format 'b' (pre-merged streams) will work without FFmpeg, though these may not offer the best available quality
- Video merging (bestvideo+bestaudio) requires FFmpeg

Q. How to fix "yt-dlp not found" error?


- App → Settings → Reset to defaults, Restart App
- Allow app to download yt-dlp automatically
- Manual solution:
 - Download yt-dlp from the [official releases page](https://github.com/yt-dlp/yt-dlp/releases)
 - Place the executable (`yt-dlp` or `yt-dlp.exe`) in the `bin/` folder
 - Restart the application

Download Performance

Q. What download speed should I set? A.

 5M (5 MBps) is recommended for most users because:

- Conservative: Unlikely to be blocked by servers
- Stable: Provides consistent download performance
- Network-friendly: Doesn't monopolize your internet connection
- Reliable: Works well with most hosting platforms

 Higher speeds (10M-20M) may work but risk:

- Server blocking: Some platforms detect and block high-speed downloads
- IP throttling: Your IP might be temporarily limited
- Adjust based on your: Internet plan, network stability, and target platform policies.

Q. Can I change the download folder? A. Yes. Go to Settings → Download Path and select a new folder. Make sure you have write permission to that location.

Q. Why does progress bar runs multiple times for a single video download? A. This is normal behavior indicating separate component downloads:

1. Video stream - Main video content
2. Audio stream - Separate audio track
3. Thumbnail - Video preview image
4. Metadata - Video information

The application downloads these components then merges them into the final file.

Q. Why does download fail when the platform is supported by yt-dlp and I can see the video in my browser? A. This is a common issue that can happen for several reasons:

Platform anti-bot measures. many platforms actively block automated downloads:

- Uses sophisticated bot detection
- May throttle downloads from certain IP ranges
- Require cookies from a logged-in session
- May need authentication or has rate limits

Solution:

1. Update yt-dlp (if Auto-Update is disabled in settings)
2. Provide browser profile path in Settings (recommended - use your actual browser session profile path)
3. Use cookies.txt file (advanced users only - be aware of security risks when sharing cookies)

Queue & Session Management

Q. Why is my queue from previous session not showing up? A. Ensure "Multisession queue support" is enabled:

1. Go to Settings
2. Enable "Multisession queue support"
3. Restart the application

Q. Why does Windows Explorer slow down with download folder? A. This can occur when partially downloaded files accumulate: Cause: With multisession support enabled, paused and ignored/skipped downloads create temporary files:

- Manually clean the download folder periodically (Remove accumulated, partially downloaded .part files, if any)
- The application includes robust cleanup logic, but manual intervention may be needed for skipped downloads

Format & Container Issues

Q. Which format should I choose? A.

- MKV: Highest quality, supports pause/resume, better for archival
- MP4: Best compatibility (phones, tablets, social media), smaller file sizes
- Single best (b): Fastest download, no merging required, but may compromise on quality

Q. What happens if my chosen format fails to download? A.

- The app automatically uses a smart fallback system to ensure your download succeeds:
- Format Fallback Chain:
 - MKV → MP4 → Single best (if MKV fails, tries MP4, then single file)
 - MP4 → Single best (if MP4 fails, tries single file)
 - Single best (no fallback needed - most reliable)
- How It Works:
 1. Starts with your preferred format selection
 2. If any issues occur (merging failures, codec problems, etc.)
 3. Automatically retries with the next format in the chain
- Example Scenarios:
 - MKV download fails → Automatically tries MP4 → Download succeeds
 - MP4 download fails → Automatically tries single file → Download succeeds
 - No manual intervention needed - the app handles it automatically

This system ensures maximum download success rates while respecting your format preferences.

Q. Why are MKV files bigger? A.

- MKV files are often larger because they are commonly used to store the highest quality video and audio available, including lossless audio tracks.
- MP4 files are more focused on broad compatibility and are typically compressed more for smaller file sizes.

Q. Why is pause/resume not supported for MP4 format like MKV? A. Technical limitation of MP4 container:

- MKV: Supports true pause/resume at any point
- MP4: The MP4 container format doesn't support efficient resuming of partial downloads like MKV does. Resuming MP4 downloads often requires re-downloading significant portions.
- MP4: Can only resume from certain checkpoints

Q. How can I pause playlist download with MP4 format as pause isn't supported (with MP4)?

A. You can pause(functionally) , with caveats (With - Multisession queue support):

- With MP4 merge format, Playlist progress is saved between sessions (exit and restart)
- Already downloaded videos will not be downloaded again when you restart downloader.
- Only last partially downloaded video (at exit time) will downloaded again to continue playlist download.

Ensure: "Multisession queue support" is enabled. The application will remember your position in the playlist and only overwrite the currently downloading video if interrupted.

Q. Why isn't thumbnail showing for MKV files? A. MKV thumbnails require external system support:

- Windows: Install Icaros (<https://www.majorgeeks.com/files/details/icaros.html>) to enable MKV thumbnail previews in File Explorer
- Linux: Most modern file managers (Nautilus, Dolphin) support MKV thumbnails natively
- Note: This is a Windows Explorer limitation, not an application issue. The thumbnails are embedded correctly in the files.

Technical Questions

Q. Where are my downloads stored? A. Default paths:

- Windows: C:\Users\[YourName]\Videos\ or app folder
- Linux: ~/Videos/ or ~/.local/share/Video Downloader/videos/
- macOS: ~/Videos/ or application directory

Q. Where are application settings stored?

- Windows: %LOCALAPPDATA%\Video Downloader\
- Linux: ~/.local/share/Video Downloader/
- macOS: ~/Library/Application Support/Video Downloader/

Q. Can I run multiple instances? A. No, only one instance can run at a time to prevent conflicts.

Q. Does it support proxy servers? A. Not currently through the GUI. Advanced users can configure yt-dlp's proxy settings manually.

Advanced Topics

Q. Why are some videos downloading slowly despite fast internet? A. Common causes:

1. Server limitations: Source platform may throttle downloads
2. Geographic restrictions: Content delivery network delays
3. Check your network settings

Q. Why do some downloads fail with "Unsupported URL"? A. Possible reasons:

1. Platform not supported by yt-dlp
2. Regional restrictions on the content
3. Private/age-restricted content requiring cookies
4. Temporary server issues on the source platform

Check: The yt-dlp supported sites list (<https://github.com/yt-dlp/yt-dlp/blob/master/supportedsites.md>) for platform compatibility.

Q. What if the app shows "Permission denied"? A.

- On Linux/macOS, make sure the download path and temp folder are writable:
- `chmod -R 755 ~/Videos`
- On Windows, run the app as administrator.

Q. How do I clean up stuck temp files? A. If the app closes unexpectedly, leftover .part files may remain. The app tries to remove them on exit, but you can safely delete them manually from the temp/ folder.

Q. Where are logs stored? A. You can find the latest session logs in the data/ folder.

Q. Can I use my own yt-dlp or FFmpeg binaries? A. Yes. Place your custom builds inside the bin/ folder with the same names (yt-dlp, ffmpeg). The app automatically detects and uses them.

Q. How do I use cookies for private videos? A. In Settings, specify the path to your cookies.txt file or enable "Cookies from Browser." This enables downloading of age-restricted or private content.

Q. I deleted desktop shortcut, how can I get it back? A. Close the Video Downloader application completely

- Windows:
 - Right-click on mt-vdl.exe
 - Select "Send to" → "Desktop (create shortcut)"
 - Or
 - Navigate to the application folder → data/
 - Delete the file called first_run.flag
 - Restart the application - it will prompt create the desktop shortcut
- If not Windows OS:
 - Navigate to the application folder → data/
 - Delete the file ./mediatools (rm ~/.mediatools)
 - Restart the application - it will prompt create the desktop shortcut

6.2 Troubleshooting

Common Issues and Solutions

Q. "yt-dlp not found" Error? A. See "Installation & Setup" section above for complete solutions.

Q. "FFmpeg not found" Error? A. Cause: FFmpeg not installed or not accessible

1. Go to App → Settings → Reset to defaults, Restart App
2. Allow app to download FFmpeg, if fails try step 3
3. Manually download FFmpeg if needed:
 - Windows: <https://www.gyan.dev/ffmpeg/builds/>
 - Linux: `sudo apt install ffmpeg`
 - macOS: `brew install ffmpeg`
 - Place in application's bin folder

Q. Download Fails Immediately? A.

Possible Causes:

1. Video no longer available
2. Private video requiring authentication
3. Unsupported platform

Solution:

1. Verify URL is correct and accessible in browser
2. Check if video requires login (use cookies)
3. Try updating yt-dlp
4. Check platform support (check yt-dlp supported domain list)

Q. Slow Download Speeds? A.

Possible Causes:

1. Internet connection speed
2. Server-side throttling
3. Download speed limit setting

Solution:

1. Check your internet speed
2. Check if download speed set in Settings too low
3. Some platforms throttle, particularly during peak hours

Q. Videos Download but Won't Play? A.

Possible Causes:

1. Incomplete download
2. Corrupted file
3. Missing codecs in media player

Solution:

1. Check file size matches expected size
2. Re-download the video
3. Try different media player (VLC/MPV/MPC-BE recommended)
4. Update your media player codecs

Q. Application Won't Start? A.

Solution:

- Check system requirements
- Run as administrator (Windows)
- Check antivirus isn't blocking
- Check if `bin/` folder has necessary executables
- Reinstall the application
- Delete data folder to reset
- Try running app executable from command window

Q. Queue Not Persisting? A. Cause: Multisession queue support disabled

Solution:

1. Go to Settings → Queue Settings
2. Enable "Multisession Queue Download Support", Save
3. Restart App
4. Queue will now save between sessions

Q. Downloads stuck at 0% or very slow? A.

Solutions:

- Check your internet connection
 - Try a different video URL to test
 - Restart the application
-

7. Contributing and Feedback

Reporting Bugs

If you encounter issues:

1. **Check This Guide:** Review [FAQ and Troubleshooting](#) section
2. **Verify Latest Version:** Update to latest release
3. **Gather Information:**
 - Operating system and version
 - Application version
 - Steps to reproduce
 - Error messages or logs
4. **Submit Issue:** <https://github.com/MediaTools-tech/mediatools/issues>

Feature Requests

Have an idea for improvement?

1. **Check Existing Issues:** Search for duplicates
 2. **Create New Issue:**
 - Clear description of the feature
 - Use case explanation
 - Expected behavior
 3. **Use Proper Label:** Apply the "enhancement" label
-

8. Privacy and Legal

8.1 Privacy and Security

Data Privacy

MediaTools Video Downloader:

- Does **NOT** collect personal information
- Does **NOT** send usage statistics
- Does **NOT** require account creation
- Does **NOT** connect to external servers (except to the video platforms for downloading content)

Cookie Security

When using cookie import:

- Cookies are stored locally only
- Never transmitted to third parties
- Encrypted in browser's own storage
- Treat cookie files as sensitive data

Safe Usage Practices

1. **Download Legal Content:** Respect copyright and terms of service
2. **Verify URLs:** Ensure links are from trusted sources
3. **Scan Downloads:** Use antivirus on downloaded files if concerned
4. **Update Regularly:** Keep application and dependencies updated
5. **Review Permissions:** Ensure application has necessary file permissions

8.2 Legal Information

License

MediaTools Video Downloader is licensed under the MIT License.

Copyright (c) 2025 MediaTools (Bala)

Permission is granted to use, modify, and distribute this software under the terms specified in the LICENSE file.

Third-Party Software

This application uses:

yt-dlp

- License: Unlicense
- Source: <https://github.com/yt-dlp/yt-dlp>
- Purpose: Video downloading engine

FFmpeg

- License: LGPL 2.1 / GPL 2.0
- Source: <https://ffmpeg.org>
- Purpose: Media processing and conversion

These components retain their original licenses.

DISCLAIMER

MediaTools Video Downloader is a graphical user interface (GUI) tool that facilitates the use of yt-dlp, an independent third-party video downloading application. MediaTools does not directly download videos; it provides a user-friendly interface to execute yt-dlp commands.

DEPENDENCY NOTICE

yt-dlp is downloaded only with explicit user permission. Users may alternatively obtain yt-dlp independently and place the executable in the application's /bin folder.

USER RESPONSIBILITIES

By using MediaTools Video Downloader, you agree that you are solely responsible for:

- Compliance with all applicable laws and regulations in your jurisdiction
- Respecting copyright, intellectual property rights, and digital rights management
- Adherence to platform terms of service and usage policies
- Obtaining proper authorization or licensing for downloaded content
- Any legal consequences resulting from the use of downloaded content

NO WARRANTY

MediaTools Video Downloader is provided "AS IS" without warranty of any kind, either express or implied, including but not limited to warranties of merchantability, fitness for a particular purpose, or non-infringement.

LIMITATION OF LIABILITY

The developer and MediaTools organization shall not be held liable for any misuse of this software, legal violations, copyright infringement, or damages arising from the use or inability to use this software.

INTENDED USE

This software is designed for downloading publicly accessible content that you have the legal right to access and download. Use for any other purpose is at your own risk.

9. Contact & Support

Email Support:

- Email: bala.lv.555@gmail.com

Website

- <https://mediatools.tech>

GitHub Issues:

- Bug reports: <https://github.com/MediaTools-tech/mediatools/issues>
- Feature requests: Same link, label as "enhancement"
- Public discussion and community help

Documentation:

- This user guide
- README.md in application folder
- GitHub wiki (when available)

Community

Stay updated and connect:

- GitHub: <https://github.com/MediaTools-tech/mediatools>
- Watch repository for updates
- Star the project if you find it useful
- Share feedback and suggestions

Credits

Developed by: Bala

Organization: MediaTools

Special Thanks:

- yt-dlp development team
 - FFmpeg project contributors
 - Open source community
 - Early testers and bug reporters
-

10. Version History

Version 1.0.0 (Initial Release)

- First stable release
 - Core download functionality
 - Queue management
 - Settings configuration
 - Cross-platform support
 - Cookie integration
 - Auto-update feature
-

Document Version: 1.0.0

Last Updated: October 2025

Document Author: Bala (MediaTools)

End of User Guide

For the latest version of this guide, visit: <https://github.com/MediaTools-tech/mediatools>

Thank you for using MediaTools Video Downloader!
