CSC207 Project Blueprint

Group: TUT0501-13

Team Name: Full marks

1. Domain

Convert YouTube videos to mp3 audios

An application that allows a user to paste a YouTube link, download the video as an MP4 file using yt-dlp, and then convert that MP4 file into an MP3 audio file using FFmpeg.

2. User Stories

1. As a user, I want to paste a YouTube video link and download it as an MP4 file so that I can watch it offline.
2. As a user, I want to convert a downloaded MP4 file into MP3 format so that I can keep only the audio for music or lectures.
3. As a user, I want to choose where the downloaded and converted files are saved so that I can easily find them later.
4. As a user, I want to see a progress indicator for downloading so that I know when each task is complete.
5. As a user, I want to paste a YouTube playlist link to download its contents so that I can quickly obtain related videos.
6. As a user, I want to choose which playlist videos to exclude from conversion so that I can download only the desired items.

3. Use Cases

3.1. Use Case 1: Download YouTube Video (MP4)

Main Flow:

- User pastes a YouTube video URL.
- System validates the URL.
- System calls `yt-dlp` (via command line) to download the video in MP4 format.

- System saves the MP4 file to the selected folder.

Alternative Flows:

- Invalid URL → show error message.
- Network failure → display retry option.

3.2. Use Case 2: Convert MP4 to MP3

Main Flow:

- User selects a downloaded MP4 file.
- System uses `FFmpeg` to extract audio and save it as MP3.
- System saves the new MP3 file in the selected folder.

Alternative Flows:

- If conversion fails, show error message.
- If an MP3 with the same name exists, ask to overwrite or rename.

3.3. Use Case 3: Choose Save Location

Main Flow:

- Before download or conversion, user selects a destination folder.
- System stores the selected path in settings.
- System saves all future files (MP4 and MP3) to this folder.

Alternative Flows:

- If the folder path is invalid or inaccessible, system prompts user to choose another location.
- If no path is selected, system uses default "Downloads" directory.

3.4. Use Case 4: Show Progress

Main Flow:

- During download, system displays progress (e.g., 0–100%).
- When complete, system shows "Download finished" or "Conversion complete" with file path.

Alternative Flows:

- If the process is canceled, stop the operation and remove partial files.

3.5. Use Case 5: Download YouTube Playlist as MP4 Files

Main Flow:

- User enters YouTube playlist URL.
- System displays names of all videos with checkboxes (all selected by default).
- User may deselect or reselect specific videos.
- System uses `yt-dlp` to download the selected videos as MP4 files.
- System saves files to the specified folder.

Alternative Flows:

- Invalid playlist URL → show error message.
- Network issue while validating URL → prompt retry or cancel.
- For unavailable or region-locked videos → show "skipped" message and continue remaining downloads.

3.6. Use Case 6: Exclude Playlist Videos from Conversion

Main Flow:

- After downloading a playlist, system shows list of all MP4 videos.
- Each video has a checkbox to include or exclude from MP3 conversion.
- User confirms selection.
- System uses `FFmpeg` to convert only the checked videos to MP3 format.

- System saves the MP3 files to the chosen folder.

Alternative Flows:

- If no videos are selected, system shows a message: "No items selected for conversion."
- If conversion for a video fails, system logs the error and continues remaining conversions.

4. MVP (Minimal Viable Product)

Lead	Use Case	User Story
Jason	Download YouTube Video (MP4)	1
Natasha	Convert MP4 to MP3	2
Mohamed	Choose Save Location	3
Daniel	Show Progress	4
Christina	Download YouTube Playlist as MP4 Files	5
Ricky	Exclude Playlist Videos from Conversion	6

5. Proposed Entities for the Domain

Entity Name	Instance Variables
DownloadTask	String url, String status, String outputPath, double progress
ConversionTask	String sourceFilePath, String targetFilePath, String status
UserSettings	String downloadDirectory, String audioDirectory, boolean autoOpenFolder

6. Proposed API for the Project

API 1: yt-dlp (Command-Line API)

Link: `https://github.com/yt-dlp/yt-dlp`

Purpose: Downloads YouTube videos as MP4 files.

Integration: Java's `ProcessBuilder` executes the command:

```
yt-dlp -f mp4 <URL>
```

Status: Successfully tested manually outside Java; integration via process execution confirmed feasible.

API 2: FFmpeg (Command-Line API)

Link: `https://github.com/yt-dlp/FFmpeg-Builds`

Purpose: Converts MP4 files into MP3 format.

Integration: Called using:

```
ffmpeg -i input.mp4 output.mp3
```

Status: Tested manually; conversion from MP4 to MP3 confirmed successful.