Detailed Requirements Document: Adapting Music to Dramatic Short Videos.

---

1. Introduction

Project Overview: The project aims to develop a system that automatically adapts music to dramatic short videos, enhancing the emotional impact and storytelling.
This involves analyzing the video, understanding their dramatic elements, identifying a constant object, analyzing its movement or facial expressions, and matching them with suitable music tracks, or providing a detailed graph depicting the object's difference across the scene.

Objectives:

- Automate the adaptation of music to fit the mood and theme of the scenes\videos.
- Provide a user-friendly interface for filmmakers and people alike, to customize music suggestions.
- Leverage AI and machine learning for scene analysis and music matching.
- Provide a detailed graph for machine learning students, etc. to help with their studies.

---

2. Identify Stakeholders:

The stakeholders for this project include:

1. **Filmmakers and Directors:** The primary end-users who want to enhance their movies' emotional impact through precise music adaptation.
2. **Composers and Music Producers:** Professionals who provide the music tracks used in the system.
3. **Cinema/Theatre Students:** Customers using this tool to help with their studies.
4. **Content Creators:** Enhance emotional impact, improve storytelling, and elevate their content's overall quality.
5. **Machine learning students**: Users that will take advantage of our platform to benefit them in video analysis studies and machine learning.

---

3. Functional Requirements

Key Features

1. **Scene Analysis**
    - Analyze short videos for visual and auditory cues such as brightness, color grading, motion intensity, and dialogue tone.
    - Detect and classify the dramatic theme (e.g. suspense, romance, action, tragedy).
    - Find and Analyze movement or facial expressions of a certain object (or animal, actor, etc.)

2. **Music Matching and Adaptation**
    - Match analyzed scenes with suitable tracks from a pre-defined music library.
3. **Custom Music Suggestions**
    - Enable users to select a preferred track from the three recommendations provided by the system.
    - Offer alternative options upon request.
4. **Machine learning graph**
    - Provide a detailed graph describing the difference of movement (or change in facial expression) and concise description of the identified object that appears throughout the scene/short video.
5. **Editing and Exporting**
    - Allow users to fine-tune music parameters (e.g., volume, fade-in/out effects – optional)
    - Export the adapted music file and scene in a compatible format.

---

4. Architectural Requirements

1. **System Architecture**
    - **Frontend:** User interface for uploading short videos, viewing analysis, receiving completed video results or graph analysis, and customizing music.
    - **Backend:** Modules for AI/ML-based scene analysis and music adaptation.
    - **Database:** Store music tracks and scene metadata. (possible on the cloud)
    - **Integration:** APIs for external music libraries and video editing tools.
2. **Performance Indicators**
    - Response time for analysis and adaptation: 15-30 seconds for small videos and up to 5 minutes for larger videos.
3. **Security**
    - Use secure protocols (e.g., HTTPS, TLS) for data transfer.

5. Technological Requirements

1. Programming Languages and Frameworks

- **Frontend:** React.js for a dynamic user interface.
- **Backend:** Node.js for server-side logic.

2. AI/ML Tools

- Use PyTorch for developing machine learning models.
- Pre-trained models for visual and audio feature extraction (e.g., OpenCV, LibROSA).

3. Database

- PostgreSQL for storing metadata and user data. (possible cloud)

4. APIs and Libraries

- Music and video processing: FFMPEG, LibROSA.
- External music libraries: Spotify API, YouTube Music API.
- Google cloud API/Azure Video Indexer usage for scene analysis and video output.

5. Hosting and Deployment

- Cloud platforms: AWS, Azure, or Google Cloud for scalability.
- CI/CD pipelines with GitHub Actions.

6. Compliance and Standards

- Ensure compliance with GDPR and CCPA for handling user data and metadata securely and ethically.
- Adhere to standards like ISO/IEC 27001 for information security management and industry guidelines for AI transparency in machine learning models.
- Follow licensing requirements for external APIs (e.g., Spotify API, YouTube Music API) and music libraries.

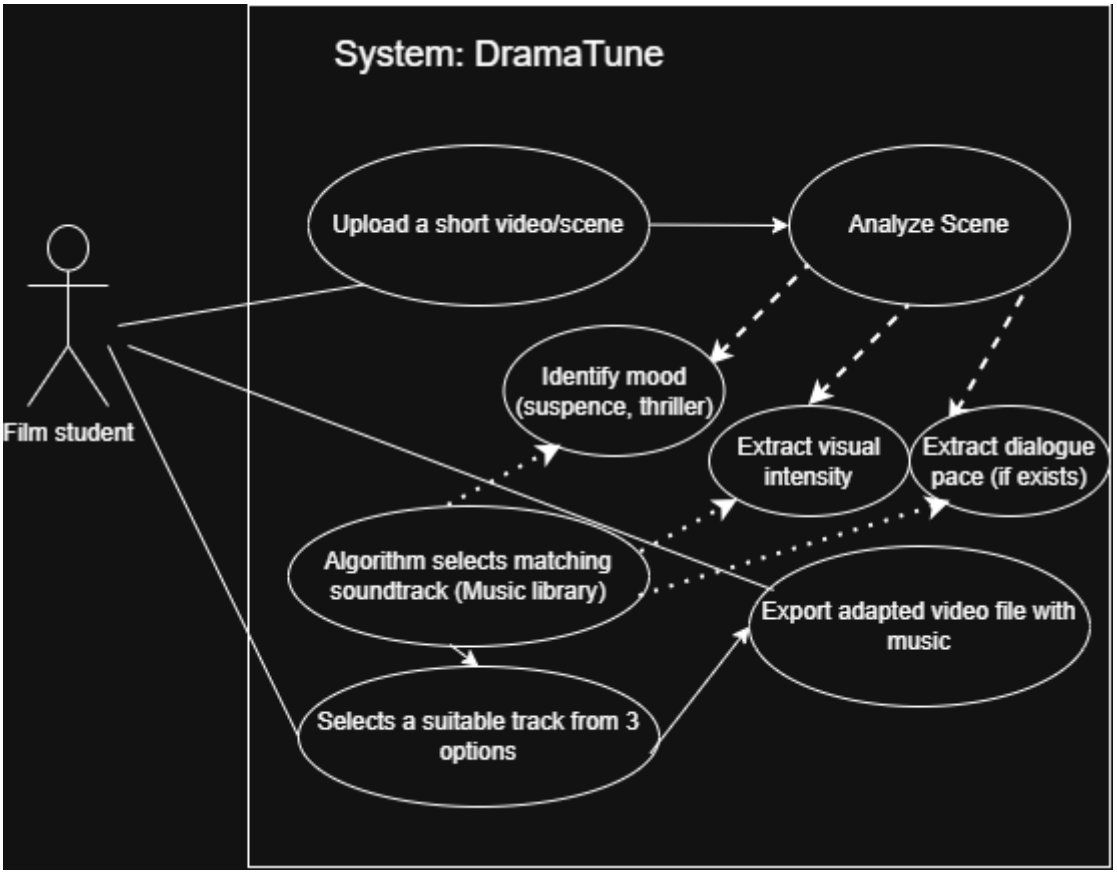
7. Testing and Quality Assurance

- Unit testing: Pytest, Jest.
- Performance testing: JMeter, Locust.
- Automated testing for critical workflows.

6. Example Use Case

Scenario: A film student uploads a 2-minute dramatic scene. The system analyzes the visual intensity and dialogue pace, identifies it as a suspenseful and tense scene, and selects a matching track from the music library.
The film student customizes the track by adjusting the tempo and exports the adapted file for integration.

Primary actor: film student/film maker/machine learning student



Basic Flow:

1. The film student uploads a 2-minute dramatic video scene.
2. The system analyses the video to extract visual intensity and dialogue pace.
3. The system classifies the scene as suspenseful and tense.
4. The system selects a matching music track from the music library.
5. The system presents the recommended track to the film student (out of 3 options to choose).
6. The film student chooses a suitable music track from the 3 options.
7. The system uploads an adapted version of the scene with the chosen music track.
8. The system exports the adapted music file.
9. The film student downloads the adapted track.

Expected Outcome: The user receives a professionally adapted music track.

Alternate Flow: The film student is not satisfied with the music track video combination

Flow deviation:

1. This alternate flow is initiated after Step 7 of the Basic Flow.
2. The film student indicates dissatisfaction with the current music–video combination.
3. The system allows the film student to request an alternative music track or modify the current track parameters (e.g., tempo).
4. The system presents an updated music preview based on the user's selection.

Return Point:

- The use case returns to Step 1 of the Basic Flow

Postcondition:

- A revised music track is available for preview and selection.

---

7. Conclusion:

The system will bridge the gap between dramatic storytelling and music composition, enabling filmmakers (and students) to create compelling cinematic experiences effortlessly.
By combining advanced AI/ML technologies with intuitive user interactions, this project will revolutionize the way music is adapted to dramatic scenes, or create a simple platform for analyzing videos by a given object (or animal, person), and provide the user with a description and graph depicting the difference and change in the person's facial expressions or the object's movement along the short video/scene.