Project Proposal

1. Project Title: <u>DoodleDJ</u>

2. Team Information

- **Team Name**: <u>Byte On</u>
- **Team:**
  - **Atharva Swami:**

    <u>Role</u>: Team Lead

    <u>Responsibilities</u>: Create and assign tickets to team members for task management and accountability. Conduct regular progress meetings to identify and address roadblocks, ensuring alignment across the team. Perform code reviews and testing to maintain quality standards and ensure functionality. Provide support to teams or individuals facing challenges to guarantee timely project delivery.

  - **Sneh Shah:**

    <u>Role</u>: Machine Learning Engineer

    <u>Responsibilities</u>: Develop and implement ML models to analyze music features for mood detection, enabling dynamic visual adaptations. Optimize real-time performance, collaborate with the backend team for seamless integration, and refine algorithms.

  - **Ravneet Kaur**:

    <u>Role</u>: Backend Developer

    <u>Responsibilities</u>: Implement server-side logic and APIs for seamless performance ensuring efficient request handling, create and manage secure database schemas and data pipelines to handle user data and artwork, and integrate ML models.

  - **Ishita Mehta:**

Role: Backend Developer

Responsibilities: Implement server-side logic and APIs for seamless performance ensuring efficient request handling, create and manage secure database schemas and data pipelines to handle user data and artwork, and integrate ML models.

- **Harsh Chheda:**

  Role: Front End Developer.

  Responsibilities: Develop an intuitive and responsive user interface, implement seamless integration with backend APIs for real-time music generation, create dynamic music visualization features, optimize rendering performance for the drawing canvas and music playback, conduct thorough testing of UI components, enhance accessibility for all users, implement feedback mechanisms for user input.

- **Rishabh Budhia:**

  Role: Front End Developer.

  Responsibilities: Develop an intuitive and responsive user interface, implement seamless integration with backend APIs for real-time music generation, create dynamic music visualization features, optimize rendering performance for the drawing canvas and music playback, conduct thorough testing of UI components, enhance accessibility for all users, implement feedback mechanisms for user input.

- **Anannya Patra:**

  Role: UI/UX Designer

  Responsibilities: Design an intuitive, responsive and visually engaging user interface that guides users through creating drawings and receiving personalized music playlists, ensuring an aesthetically pleasing and user-friendly experience. Identify user needs, design interaction models and continuously test and refine the design to ensure it meets user expectations.

- **Sai Varnitha Reddy:**

  Role: UI/UX Designer

  Responsibilities: Design an intuitive, responsive and visually engaging user interface that guides users through creating drawings and receiving personalized

music playlists, ensuring an aesthetically pleasing and user-friendly experience. Identify user needs, design interaction models and continuously test and refine the design to ensure it meets user expectations.

- **Shalin Shah**:

  Role: Machine Learning Engineer

  Responsibilities: Develop and implement ML models to analyze music features for mood detection, enabling dynamic visual adaptations. Optimize real-time performance, collaborate with frontend teams for seamless integration, and refine algorithms.

3. Problem Statement

- **Description**: DoodleDJ is an innovative application that curates personalized music playlists in real time, responding as a user draws or sketches. While the user's strokes come to life on a digital canvas, the application analyzes various aspects of their artwork — such as texture, color blending, cultural elements, and form—to create and adjust a dynamic playlist. This playlist enhances the user's creative flow and matches their mood. By merging art and music, DoodleDJ offers an immersive experience that stimulates auditory and visual senses, inspiring and boosting creativity.
- **Significance**: The issue at hand is significant because it directly impacts the creative expression of many individuals. Music plays a crucial role in fostering artistic flow and enhancing creativity, yet the time-consuming task of manually curating playlists can disrupt this vital process. This challenge affects a diverse group of users, including millions of music enthusiasts, artists, and casual listeners. Without a platform that seamlessly integrates music exploration with visual creativity, these individuals face limitations in conveying emotions through art and sound. Addressing this issue is vital to support and empower their creative journeys.
- **Current Challenges**: Many users rely on music to boost their creativity and enter a "flow" state while drawing or painting. However, current solutions lack real-time adaptations that respond to the specific characteristics of the artwork. Typically, available options fall into two categories: dedicated music streaming services or separate drawing applications. Unfortunately, these do not provide a cohesive, interactive, and personalized experience that effectively integrates both elements.

4. Proposed Solution

- **Overview**: The proposed solution is a web application called DoodleDJ, which connects music and art by dynamically curating music playlists in real-time. It integrates with a

digital drawing platform, analyzing the user's brush strokes, color choices, and other visual elements. This creates a cohesive and personalized music experience that evolves as the artwork progresses.

- **Key Features**:
  - **Real-Time Music Curation:** Analyze the characteristics of the artwork, such as colors, strokes, and patterns, to generate a matching music playlist in real-time.
  - **Mood-Based Adaptation:** Detect the emotional tone of the user based on the artwork and adjust the playlist accordingly.
  - **AI-Driven Personalization:** Utilize machine learning models to tailor music suggestions to the user's preferences and past behavior.
  - **Seamless Integration:** Provide an intuitive interface that integrates with popular digital art tools.
  - **Feedback Loop:** Allow users to provide input on playlist adjustments, which will improve personalization over time.
  - **Cultural and Stylistic Sensitivity:** Incorporate elements like cultural motifs and artistic styles to enhance music choices.
  - **Database:** Users should be able to save their artwork, which the AI models can use to create better music recommendations.
- **Value Proposition**: DoodleDJ improves upon existing options by providing an interactive, unified platform where music and art seamlessly come together. Unlike static playlists or separate applications, DoodleDJ actively responds to the user's creative process, enhancing focus and inspiration. Its real-time adaptability and personalized suggestions create a unique experience tailored to each individual user. This innovative approach transforms both music streaming and art creation, fostering deeper emotional expression and unlocking new creative possibilities.

5. Target Audience

• **Who Will Use It?**:

  ❖ Professional and amateur artists.
  ❖ Digital creators using tools like Photoshop, Procreate, or Figma.
  ❖ Music Enthusiasts
  ❖ Hobbyists and Casual Creators
  ❖ Students and Learners
  ❖ General Users

• **User Needs**:

- ❖ Real-time music adaptation based on the mood of the artwork
- ❖ Personalized music playlists tailored to user preferences and creative styles
- ❖ Seamless integration of art and music to maintain creative flow without interruptions
- ❖ A unified platform to combine music discovery and artistic expression
- ❖ Easy-to-use interface that minimizes distractions, enhancing focus and creativity
- ❖ Introduction to new genres, styles, and artists to inspire creativity and exploration

6. Project Scope

- **In Scope**: What will be included in the project
  - Create a web application that adapts music in real time based on dynamic visual inputs.
  - Integrate the Spotify API for music streaming and feature extraction.
  - Implement machine learning models to detect mood and enable real-time interactions.
  - Design an interactive, user-friendly drawing canvas.
  - Build robust error handling for API integration and model interactions to ensure a smooth user experience
  - Generate recommendations based on previous artworks stored in the database using ML.
- **Out of Scope**: Offline functionality, mobile app development

7. Technology Stack

- **Frontend**: ReactJS, HTML, CSS, JavaScript, Bootstrap, Styled Components
- **Backend**: Flask, Python, Fast API
- **Database**: MongoDB
- **Machine Learning:** Python, PyTorch, TensorFlow, Open Source LLM Models, Ollama, Librosa (for audio processing)
- **Additional Tools:** GitHub (version control), Figma (UI/UX design), Balsamiq Wireframe (prototyping), ClickUp (project management)

8. Development Plan

- **Methodology**: Kanban
- **Sprint Goals**:

- ○ **Sprint 1:** Gather detailed requirements for the prototype and create the product backlog, ensuring alignment with the DoodleDJ concept.
- ○ **Sprint 2:** Conduct a thorough risk analysis and feasibility study, evaluating technical, design, and user-interaction challenges.
- ○ **Sprint 3:** Set up a Git repository for version control and complete the initial project review.
- ○ **Sprint 4:** Finalize the architecture design and develop visual mockups to demonstrate the application's functionality and user experience.
- ○ **Sprint 5:** Start implementing the features by refining backlog items and setting up a foundational code structure.
- ○ **Sprint 6:** Deliver the first feature prototype, focusing on real-time playlist adaptation based on color and brushstroke detection.
- ○ **Sprint 7:** Resolving bugs and issues, playlist synchronization.
- **Timeline**:
  - ○ **Weeks 1–4:** Focusing on planning, risk evaluation, designing wireframes and Git repository setup to establish a strong foundation for development.
  - ○ **Weeks 5–8:** Finalizing architecture design, mockup creation, and start implementing the features.
  - ○ **Weeks 9–12:** Delivering the first functional prototype, resolving existing bugs, and making improvements.

9. Risks and Challenges

- **Technical Complexity:** Ensuring real-time synchronization between music and visual changes.
  - ○ Mitigation Strategy: Use efficient algorithms and test extensively to maintain synchronization accuracy.
- **API Limitations:** Dependence on third-party APIs like Spotify, which may impose restrictions or change functionality.
  - ○ Mitigation Strategy: Regularly monitor API updates and maintain fallback mechanisms for critical functions.
- **Performance Issues:** Handling low-latency requirements for seamless user interaction.
  - ○ Mitigation Strategy: Optimize backend architecture and utilize caching to reduce response times.
- **Cross-Platform Compatibility:** Ensuring the application functions consistently across different devices and browsers.
  - ○ Mitigation Strategy: Conduct rigorous testing across multiple platforms and use responsive design principles.
- **Scalability:** Managing increased user loads without affecting performance.

- ○ <u>Mitigation Strategy</u>: Leverage cloud infrastructure and implement auto-scaling to handle traffic spikes.
- **Data Privacy:** Protecting user data and complying with privacy regulations.
  - ○ <u>Mitigation Strategy</u>: Implement strong encryption and adhere to GDPR or similar compliance standards.

10. Expected Deliverables

- Minimum Viable Product (MVP): A functional web application utilizing the Spotify API, designed with a mood-driven drawing canvas, and featuring real-time synchronization that allows music to play and change dynamically in response to the drawing.

- Prototypes: Early iterations of the user interface showcasing real-time detection of mood, drawing tools, and music adjustment based on user interactions.

- Documentation: Comprehensive technical guides outlining API integrations, machine learning models, real-time synchronization processes, and the overall architecture of the system.

- Machine Learning Models: Algorithms created to analyze the mood and characteristics of music in real-time, adjusting dynamically according to drawing inputs.

- Deployed Application: A live version of the application accessible to users, offering real-time interaction between drawing modifications and music playback.

- Test Results: Reports that present performance metrics, user feedback, test outcomes, and the effectiveness of the real-time music adaptation.

- Design Assets: UX/UI mockups, wireframes, and adaptable visuals designed for smooth integration of mood detection and real-time synchronization of music.

11. Success Criteria

- Successful real-time artwork analysis
- Accurate mood-matching in playlist generation
- Seamless music playback
- Positive user feedback on creative flow enhancement

<u>Note: The sentences in this document were reviewed and rewritten using ChatGPT to ensure they are clear, concise, and free of grammatical errors.</u>