Project Title:

Audio2Art: Transforming voice prompts into visual creations using transformers

Team Name:

Team Code_Crusaders

Team Members:

- P. Jahnavi
- R. Sharanya
- Nausheen Haque
- T. Sanketha
- B. Nandhini Reddy

Phase-1: Brainstorming & Ideation

Objective:

To Develop an AI-powered platform that seamlessly converts audio files into visually captivating artwork, which empowers creators, musicians, and artists to visualize their sounds in innovative ways, enhancing creativity and self-expression.

Key Points:

1. **Problem Statement:**
    - Audio2Art is an advanced project powered by cutting-edge AI technology and transformer models designed to convert audio prompts into stunning visual representations.
    - This innovative system bridges the gap between auditory and visual experiences, providing users with the ability to generate images from voice descriptions effortlessly.
    - Audio2Art is versatile and can be utilized across various scenarios, offering creative solutions tailored to different user needs.
2. **Proposed Solution:**

- Audio2Art allows users to generate unique artworks by simply providing a voice description.
- Leveraging the power of transformers, Audio2Art can interpret complex language and context to produce detailed and accurate images.

3. **Target Users:**

   - **Artists & Designers** Generate unique visuals inspired by sound for digital and print media.
   - **Musicians & Producers** Create stunning artwork for album covers, music videos, and promotional content.
   - **Podcasters & Voice Artists** Turn spoken words into artistic visuals for branding and marketing.
   - **Content Creators & Streamers** Enhance videos and live streams with AI-generated sound-based visuals.
   - **Event Organizers & DJs** Design immersive visuals that sync with live performances.

4. **Expected Outcome:**

   - **AI-Generated Visuals from Audio** A seamless platform that transforms sound into unique, high-quality artwork.

   - **Creative & User-Friendly Experience** An intuitive interface enabling artists, musicians, and creators to generate stunning visuals effortlessly.

Phase-2: Requirement Analysis

Objective:

To Define the technical and functional requirements for the Audio2Art.

Key Points:

1. **Technical Requirements:**

   - Programming Language: **Python**
   - Backend: **Diffusion for Image Recognition, Web Speech API**
   - Frontend: **HTML, CSS, JavaScript**

2. **Functional Requirements:**

   - **AI-Based Artwork Generation** The system converts sound into visually appealing art.
   - **Multiple Art Styles** Users can select different artistic styles for customization.

- **User Dashboard** Login, track generated artworks, and download high-resolution images.
- **Share & Export Options** Allow users to download or share generated artwork.

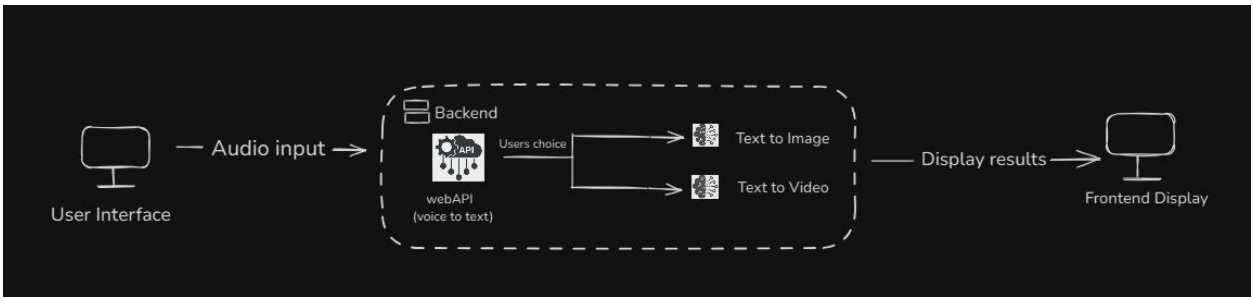
3. **Constraints & Challenges:**

   - **Real-Time Processing** – Achieving fast AI inference without performance issues.
   - **Quality & Accuracy** – Ensuring AI-generated visuals align meaningfully with the audio.
   - **Scalability** – Handling large audio files and multiple users without lag.

Phase-3: Project Design

Objective:

To Develop the architecture and user flow of the application.



Key Points:

1. **System Architecture:**

   - User records an audio via the UI.
   - The backend processes the audio and extracts features (spectrogram, frequency, etc.).
   - AI model generates artwork based on extracted audio patterns.
   - The frontend displays the AI-generated artwork

2. **User Flow:**

   - Step 1: User uploads an audio file and selects an art style.
   - Step 2: The backend processes the audio, extracts features and sends it to the AI model.
   - Step 3: The AI model generates a unique artwork based on the audio input.

- Step 4: The frontend displays the AI-generated artwork with customization options.
- Step 5: User can download, share, or save the artwork to their profile.

3. **UI/UX Considerations:**

- **Minimalist, AI-driven design** A clean, intuitive interface for effortless navigation.
- **Real-time preview** Instant display of AI-generated artwork.
- **Dark & light mode** Enhanced user experience for different preferences.

Phase-4: Project Planning (Agile Methodologies)

Objective:

Break down development tasks for efficient completion.

Sprint	Task	Priority	Duration	Deadline	Assigned To	Dependencies	Expected Outcome
Sprint 1	Audio to Text conversion	🔴 Medium	2 hours (Day 1)	End of Day 1	Nandhini & Sanketha	Web-speech-api, SpeechRecognition	Convert audio input into text
Sprint 1	Frontend UI Development	🟡 Medium	2 hours (Day 1)	End of Day 1	Nausheen & Nandhini	Html,css , javascript	Basic UI with input fields
Sprint 2	Text to Image & Video generation	🔴 High	5 hours (Day 2)	Mid-Day 1	Sharanya & Jahnavi	Diffusion and Stable Diffusion	Search functionality with filters
Sprint 2	Error Handling & Debugging	🔴 High	1.5 hours (Day 2)	End of-Day 2	Sharanya & Jahnavi	UI inputs	Improved API stability
Sprint 3	Testing&UI Enhancements	🟡 Medium	1.5 hours (Day 2)	Mid-Day 2	Sanketha & Nausheen	Interface response,UI layout completed	Responsive UI, better user experience
Sprint 3	Final Presentation & Deployment	🟢 Low	1 hour (Day 2)	End of Day 2	Entire Team	Working prototype	Demo-ready project

Sprint Planning with Priorities

Sprint 1 – Audio to Text conversion (Day 1)

(🔴 **Medium Priority)** SpeechRecognition
(🔴 **High Priority)** Integrate **Web-speech-api,**
(🟡 **Medium Priority)** Build a **basic UI with input fields**.

Sprint 2 – Core Features & Debugging (Day 2)

(🔴 **High Priority)** Implement models for **text to image &video** generation.
(🔴 **High Priority)** Debug API issues & handle **integration issues**.

Sprint 3 – Testing, Enhancements & Submission (Day 2)

(🟡 **Medium Priority)** Test responses, refine UI, & fix UI bugs.
(🟢 **Low Priority)** Final **demo preparation & deployment**.

Phase-5: Project Development

Objective:

To Implement core features of the Audio2Art.

Key Points:

1. **Technology Stack Used:**
    - **Frontend:** HTML, CSS, JavaScript
    - **Backend:** Diffusion for Image Recognition, Web Speech API
    - **Programming Language:** Python
2. **Development Process:**
    - **Build AI & Backend** Implement AI models for audio-to-art conversion
    - **Develop Frontend & Features** Design an intuitive UI , enable multiple art styles, and implement user authentication & dashboard.
    - **Optimize & Deploy** Enhance AI processing speed, add download/sharing options
3. **Challenges & Fixes:**
    - **Challenge:** Real-Time Processing Delays.
      Fix: Optimize AI models, use GPU acceleration, and implement efficient caching mechanisms.

- **Challenge:** Ensuring High-Quality & Relevant Artwork.
  Fix: Train AI on diverse datasets, fine-tune models, and allow user style selection for better customization.

Phase-6: Functional & Performance Testing

Objective:

Ensure that the AutoSage App works as expected.

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	Convert an audio clip into an image	The system should generate a relavent image	✅ Passed	Sanketha
TC-002	Functional Testing	Process different audio formats (MP3,WAV,OGG)	All supported formats should be processed	✅ Passed	Nausheen
TC-003	Performance Testing	Image generation time under 5 seconds	The system generate image quickly.	⚠ Needs Optimization	Nandhini
TC-004	Bug Fixes & Improvements	Fix incorrect image outputs for noisy audio..	The generated image should match the audio theme.	✅ Fixed	Jahnavi
TC-005	Final Validation	Ensure UI works on both mobile &desktop	UI should be responsive.	❌ Failed - UI broken on mobile	Sharanya
TC-006	Deployment Testing	Deploy the app with a text-to-video feature	The app should be accessible online	🚀 In Progress	DevOps

---