C4GT DMP PROPOSAL

Name	Pushpit Kamboj
Email ID	pushpitkamboj@gmail.com
Phone Number	+91 95681818377
GitHub ID	https://github.com/pushpitkamboj
Discord ID	pushpit_43794
Current occupation	Student
Education Details	Thapar Institute of Engineering and Technology Computer Engineering
Technical skills with level	JavaScript – Intermediate GenAI – Intermediate Rest API – Expert Fast API - Intermediate Devops - Novice

Title: Revolutionizing the learning experience for young minds, we're taking Music Blocks to the next level, powered by generative AI music generation. Unlocking creativity and education among students combining music with coding.

Summary

Text-to-Music Generation

- Users can input creative prompts (e.g., *"a relaxing ambient melody with piano and rain sounds"*) and the system will generate unique music samples accordingly.

AI-Driven Music Tweaking

- Users can upload or select a music sample and provide instructions such as *"make it faster"*, *"add lo-fi elements"*, or *"convert to minor scale"*. The model will transform the input audio as per the request.

Integration into Music Blocks

- The AI-generated or transformed music will be made accessible through Music Blocks' intuitive visual interface, allowing learners to integrate the generated sample into their code.

Project Detail

1. **Project Overview:**
   a. **Understanding of the project**

Based on the requirements of our project—which combines coding with music theory. We need an open-source AI music generation model capable of converting text prompts into audio, specifically focusing on instrumental outputs. The ideal model should understand and generate musically accurate elements such as tone, pitch, tempo, timbre, and other musical parameters to maintain technical and auditory fidelity.
Once an appropriate model is selected, the next step will be to integrate with sample widgets in our existing Music Blocks application, the generated sample then can be used to user's project ensuring seamless functionality and user experience.

b. Challenges Identified

One of the primary challenges in implementing AI music generation lies in the infrastructure requirements of open-source models such as MusicGen, Riffusion, and Jukebox. These models typically demand high-performance GPUs for hosting and fine-tuning, resulting in substantial financial costs related to hardware or cloud-based services. Additionally, managing the MLOps lifecycle—from deployment to scaling—requires significant technical expertise and engineering effort, making the process resource-intensive for small teams or limited budgets.

Another constraint arises from the licensing models of proprietary platforms like AIVA, Mubert, and Soundraw. While these services offer robust and high-quality text-to-music conversion, they are often locked behind subscription-based pricing models. Furthermore, restrictive licensing terms may prevent seamless integration into custom or open-source applications, limiting flexibility for developers seeking scalable and legal solutions for widespread deployment.

Lastly, latency becomes a critical issue when generating longer audio samples, especially with complex or high-fidelity models. In such cases, users may experience significant delays due to the model's inference time and the computational load on backend servers. This latency can negatively impact user experience, particularly in real-time or interactive environments, where immediate feedback is crucial.

c. Proposed Solution

After an extensive analysis of available tools and platforms, I propose utilizing Replicate—a platform that hosts open-source AI models and offers via pay-as-you-go APIs. Key benefits include:

- No need to self-host or provision expensive GPUs
- Scalable infrastructure managed by Replicate, reducing DevOps overhead
- Access to a many other community-trained models (like MusicGen, Riffusion, etc.) which we can leverage and can create multimodal backend.

This approach provides a cost-effective, scalable, and technically feasible pathway for integrating AI-generated music into our application without compromising on quality or agility.

Latency can be mitigated through intelligent query routing, directing prompts to model variants based on complexity of prompt.
I also created a sample demo of the feature
https://github.com/pushpitkamboj/AI-musicAgent

running video of the project
https://drive.google.com/file/d/1DUa5lpAOMWxy92TlXXB6YeKi46do2tL0/view?usp=sharing

Macro Implementation Details with Timelines

Time Frame	Start date	End date	Task
Phase 1	1 June	7 June	Community Engagement and discussion with mentors
	8 June	14 June	Finalizing the AI music model(s) to go on with
	15 June	28 June	A Fast API backend to handle user prompts, accept audio samples (MP3 or URL), and forward requests to the AI music model for generation.
	29 June	5 July	Add rate limiting and other security preventions to avoid abuse of API
Phase 2	7 July	12 July	Write test cases for the API
	13 June	19 June	Design the UI component which is in sync with existing interface of music blocks
	20 June	31 July	Create the frontend component and integrate it with Fast API
Phase 3	1 August	10 August	Resolve bugs and implemented additional features based on mentor's suggestions.
	11 August	24 August	Documenting the progress
	25 August	31 August	Final submission of project and mentor feedback

Availability

Number of hours available to dedicate to this project per week	20- 30 hours
Do you have any other engagements that will require your time? (projects/internships)	NO

I am not doing any internship or any other major project in the said time period and will be completely dedicated for the DMP project.

Personal Information

My name is Pushpit Kamboj, a third-year Computer Science undergraduate from Thapar University Patiala with a strong interest in software development and emerging technologies. In the past few months, I have developed a keen interest in Generative AI and its potential to drive innovation across industries. This has led me to work on several projects involving large language models, vector databases, and embedding-based retrieval systems (RAGs), where I focus on applying AI to solve real-world problems.

When I'm not wrestling with code, I probably be around smashing ping pong balls or stealing the spotlight on stage — turns out debugging and dramatic monologues go surprisingly well together.

My motivation for the project

The core motivation behind this project was deeply personal — the idea of teaching music through the Music Blocks platform resonated with a part of me I've carried since childhood. Growing up, I was captivated by EDM and electronic music, spending hours creating beats on whatever software I could get my hands on. That passion never left me.

Now, getting the chance to build something that could spark that same joy in students — to empower them to explore music creatively — feels incredibly fulfilling. Just knowing that a feature I wrote might light up a student's face, inspire their curiosity, or ignite their love for music… that's the kind of impact that makes every late night of coding worth it.

Contributions

Link to Issue	Resolution description in short	Link to pull request
https://github.com/sugarlabs/www-v2/issues/90	Categorized the FAQ questions on the website of sugarlabs.	https://github.com/sugarlabs/www-v2/pull/125
https://github.com/sugarlabs/musicblocks/issues/4626	Tried to optimize the import MIDI file function using chunking and asynchronous calling, but could not achieve significant progress so closed the PR, the issue too was closed later.	https://github.com/sugarlabs/musicblocks/pull/4647
https://github.com/sugarlabs/musicblocks/issues/4652	When running the command npx jest --coverage, unnecessary files in the coverage folder were getting tracked by the git repository. This created a large number of files inside coverage folder, coverage folder has to be in gitignore.	
https://github.com/sugarlabs/musicblocks-v4/issues/433	While cloning the music blocks v4 I got the error which depicts that docker file requires python to install some dependencies.	

CODE 4 GovTech
Building Digital Public Goods

Previous experience

Project Name	Project Description	Links
AI music agent	This project is a web application that generates original music from text descriptions using Meta's MusicGen AI model through the Replicate API. Users can enter text prompts like "electronic music" or "peaceful birds" and the app creates corresponding audio files that can be played in the browser.	https://github.com/pushpitkamboj/AI-musicAgent
AI coding assistant	I tried to reverse engineer how bolt works and tried creating my own terminal-based AI agent	https://github.com/pushpitkamboj/coding-AI-agent
Persona based AI agent	Created an AI agent of a tech creator and deployed it through Twilio, use it by sending message on WhatsApp no. +1 (415) 523-8886. send join life-ocean to activate Twilio.	https://github.com/pushpitkamboj/hitesh-chaudhary-bot
Blog application	Created a blog application where u can post the image title and content of a blog, see all posts and basic other curd operations, images of blogs are stored on S3.	https://github.com/pushpitkamboj/blog-app
Zendola organization world accessible map	Implemented the path navigation feature using overpass API. it fetches the geographical features such as mountain ranges, water, woods etc around the radius of pointer	https://github.com/zendalona/world-map-explorer-v2/pull/19