Project Proposal

MUSIC PLAYER - VISUALIZER

- Full Name: *N. Siddhi Vinayak Narisetty*
- Roll Number: *B24BB1036*
- Short Bio :

  I'm a first-year bioengineering student at IIT-Jodhpur. The world of web development and open source is quite new to me but I'm still exploring the overall tech domain so I hope this is something that clicks for me as a personal area of interest.
- Contacts :
  - Email: b24bb1036@iitj.ac.in
  - GitHub Username: siddhivn
  - Discord Username: siddhi.v.n
  - WhatsApp no : 9985467733
- Why are you interested in working on this project?

  The concept of an MPV is overall just very cool and this one specifically has features which cater to every need of an avid music listener. So being a part of building something like this would be a wonderful opportunity to not only make something awesome but also build up my web development skills as a new-to-the-field developer.

Project Understanding

- **What problem the project solves and who faces that problem?**

  Research shows that although there are a number of music visualization platforms (eg : Synthesia, Plane9, Resolume etc.) which cater to similar requirements as this one the following problems are faced in most of them -

  1. Most MPVs are not compatible with macOS and Linux. This prevents the users of these operating systems from using such softwares. This project aims to make the application platform universal, so users of all operating systems can use it without any issues.
  2. Both real time generation and hardware optimization simultaneously are absent in a number of MPVs. Users with slow and less storage systems dont allow for real time generation. This project solves that issue by ensuring the visual is generated on the spot which allows for dynamic customization while also ensuring hardware optimization i.e. irrespective of the system used it will be modified accordingly in order to prevent the issue of buffer or lag in the visual.
  3. Dynamic Customization i.e allowing the user to tweak and play with the visuals depending on their mood and overall giving a more interactive experience.
  4. Accessibility to the differently-abled i.e. high contrast option to make it easier for the visually impaired and ensuring that all features are accessible via keyboard for those with limited mobility.

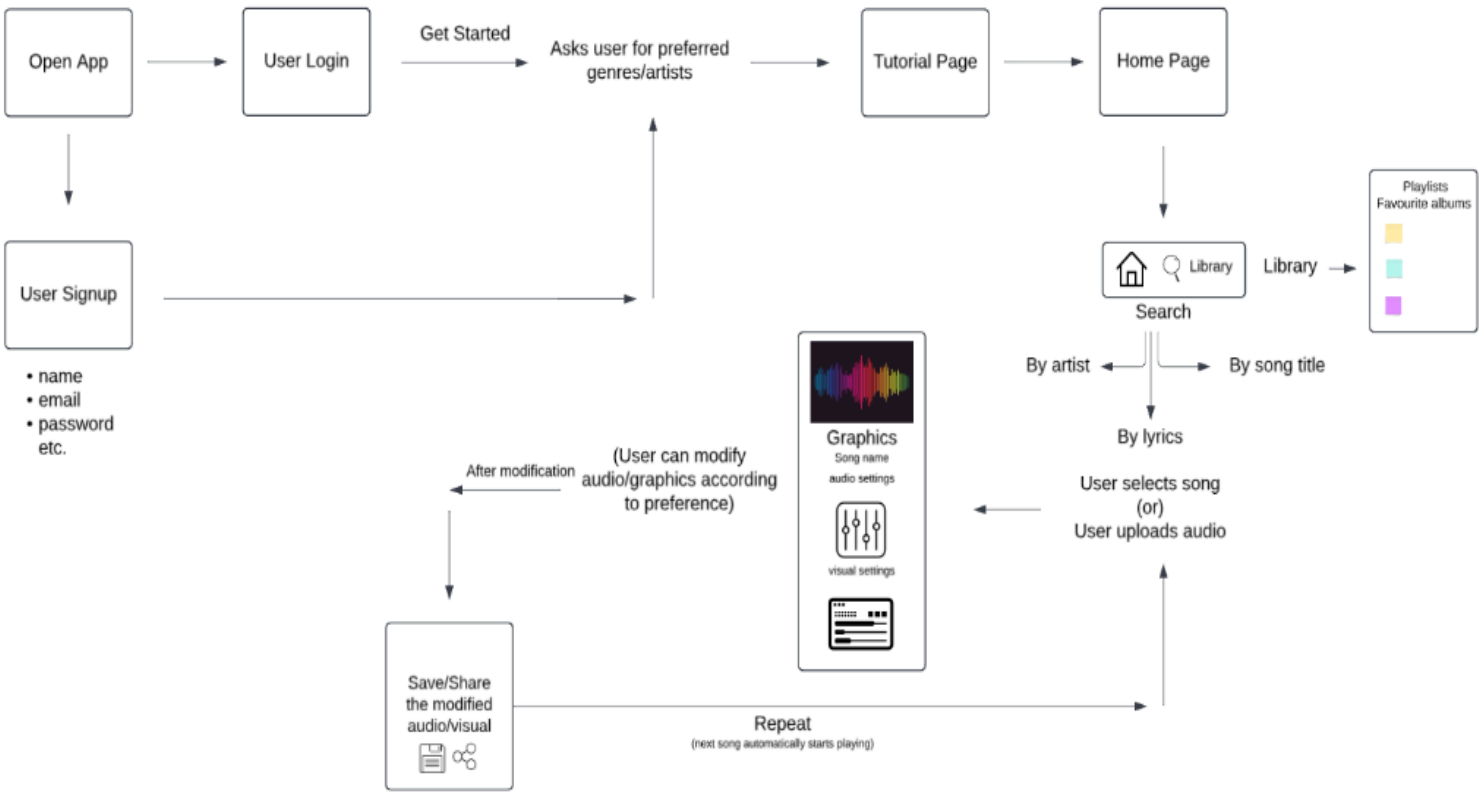
- **What do you plan to do with the project (list of features, user flow)?**

  Features :

  - Procedural generation of visuals to give users a creative, immersive experience.
  - Adjusting graphics and visuals based on the operating system and hardware.
  - Allows the user to download and upload a range of audio formats like MP3, WAV, AAC etc.
  - Deep playlist management (enhancing user experience by offering visually appealing tools for playlist management , drag and drop option for playlist management , incorporating album covers into the interface to make it more visually pleasing etc.)

- Dynamic customization (as mentioned above) i.e allowing the user to modify visuals as they are generated.
- Using SQLite to store and organize user data without the requirement of a huge server.
- Gapless playback and smooth transitions between tracks
- Full access to equalizer settings to allow the user to personalise audio according to their preference and also download and save modified audios.
- Customizable Layouts
- Integration with streaming services like Spotify and Apple Musicnand synced lyrics.

User Flow :



- **What is your tentative timeline for the progress of the project?**

  Week 1 - Project planning

  Week 2 - UI interface and initial phase of frontend development

  Week 3 - Backend development and integration of audio processing

  Week 4,5 - Final phase of backend and frontend development

  Week 6 - Debugging and testing of the project

- **Any open source alternatives to the project?**

  - Instead of Material-UI, **Bootstrap** can be used. It has numerous predefined components and styles which can be directly implemented.

  - **Tauri** can be implemented over Electron.js. It takes up relatively less storage space and has a faster run time ensuring smooth running of the application and a seamless user experience.

Technical Understanding

- **Suggest some tech stack options which you think we can use in this project?**

  1. **Real time audio processing :** Web audio API and FFT

     Web audio API for efficient audio analysis and FFT is used to extract the real time amplitude and frequency data

  2. **Hardware Optimization :** OpenGL for GPU acceleration

     OpenGL used to offload tasks to the GPU and GLSL shading language is used in writing custom shaders for visual effects.

  3. **Graphics (Frontend) :** Three.js to ensure that the visualizer draws everything efficiently.

Reusing shapes, colors, animations wherever possible to optimize texture usage and reduce rendering workload.

4. **Cross-Platform Compatibility (Frontend) :** Electron.js

   The MPV should run on all operating systems.

5. **Backend :** Node.js

   Fast efficient handling of audio processing, sync with visuals and communication with the frontend tech stack.

- **Which components of the tech stack do you know about?**

  Basic understanding of HTML and CSS, and on research, I now have a brief idea of what each component of the tech stack is used for, but overall little to no prior knowledge of the actual working or code of any of them. But I am willing to learn and eager to develop my understanding of web development.

Time Commitment

How much time (in hours per week) will you be able to commit to this project? Also mention any time phases when you will have other commitments (for example, end sems) and cannot devote the same time compared to your regular schedule.

- Approximately 10 hours per week.
- Planning on working on a robotics project for prometeo so around Jan 15-19 might be a little busy.
- Minor Exams (20-22 Feb)
- Major Exams (22-28 Apr)