PROJECT TITLE: MUSIC RECOMMENDATION APP

- **Authors**: Maximov Ilya, Zakirov Karim
- **University**: Innopolis University
- **Date**: 12 May

Background and Relevance

- The popularity of music streaming apps has made discovering new music a crucial aspect of user experience.
- Personalized music recommendations improve user engagement and satisfaction.
- Music Recommendation App leverages AI to recommend songs based on user preferences and mood.

Main Goal:

- To develop an application that recommends personalized music to users based on their feedback and mood.

Specific Objectives:

- Provide a simple, intuitive interface for users to interact with.
- Implement an AI model for personalized music recommendations.
- Enable real-time updates of recommendations as users provide feedback (like/dislike)

Technology Stack

- **Frontend**:
- Flutter
- **Backend**:
- Node.js with Express.js
- postgresql
- RESTful API

UI Design Principles

- **Minimalism**: All UI elements, such as buttons and song cards, are easy to understand and use, with no unnecessary elements.
- **Interactivity**: Every interaction with UI elements (like adding a song to favorites) provides an immediate visual feedback.
- **Responsiveness**: The interface adapts automatically to different screen sizes (mobile devices and tablets).

HAII Principles

- **Simplicity**: The UI is minimalist, and users can easily find all necessary functions, such as adding songs to favorites.
- **Feedback**: When adding or removing songs from favorites, the system immediately reflects the changes.
- **Predictability**: The system's behavior is consistent — after clicking "add to favorites," the system instantly updates the song's status.
- **Direct Manipulation**: Users directly interact with song cards and buttons to manage their content, without the need for complex menus or transitions.

Live Demonstration

Challenges

- **API Integration:**
  Ensuring smooth integration between frontend and backend, particularly when handling user preferences and recommendations in real-time.
- **State Management:**
  Managing state effectively with multiple asynchronous actions, especially when fetching user data, mood selection, and song recommendations.
- **Error Handling:**
  Handling network issues or server downtime gracefully by showing appropriate feedback to users.

Challenges

- **Mobile Compatibility:**
  Testing and optimizing the app for various screen sizes and resolutions to ensure a consistent user experience.
- **User Authentication:**
  Ensuring secure and reliable user authentication, including managing login and registration without losing the session.

Future Work

- **Enhanced Recommendation Algorithm:** Improve the recommendation engine to use machine learning models that better understand user preferences and mood.
- **Offline Mode:** Implement offline support, allowing users to access their favorite songs and recommendations even without an internet connection.
- **Social Features:** Add social features such as the ability to share playlists or favorite songs with friends.

THANK YOU FOR YOUR ATTENTION