Moodmix Development Installation Guide

1. Introduction

This document provides detailed instructions on how to install and set up Moodmix. Follow the steps carefully to ensure a successful installation.

---

2. Prerequisites

Before beginning the installation, ensure that your environment meets the following requirements:

2.1 Hardware Requirements

- **CPU: 2.0 GHz dual-core processor or better**
- **RAM: 8 GB minimum (16 GB recommended)**
- **Storage: 2 GB of free disk space (SSD recommended)**
- **Network: Broadband internet connection**

2.2 Software Requirements

- **Operating System:**
  - **Windows 10 or later (64-bit)**
  - **macOS 10.15 (Catalina) or later**
  - **Linux (Ubuntu 18.04 or later, CentOS 7 or later)**
- **Java: Java SE Development Kit (JDK) 11 or later**
- **Python: Python 3.12 or later**
- **Android Studio**
- **Flutter: Version 3.22**
- **Python Packages: flask, sounddevice, numpy, pydub, whisper, textblob**

---

3. Software Installation Guide

This section provides all the necessary steps to set up the required software.

3.1 Python Installation

1. **Install Python Packages**
   - **Run the following command to install the required Python packages:**

   ```
   pip install flask sounddevice numpy pydub whisper textblob
   ```

2. **Install FFmpeg**
   - **Follow the guide to install FFmpeg on your system:**

            - FFmpeg Installation Guide

3.2 Flutter Installation

1. Install Flutter Packages
    - Navigate to your Flutter project directory and run:

```
flutter pub get
```

3.3 Set Up Java

- Follow the guide to install Java SE Development Kit (JDK) 11 or later:
    - Java Installation Guide

3.4 Set Up Python

- Follow the guide to install Python 3.12 or later:
    - Python Installation Guide

3.5 Set Up Android Studio

- Follow the guide to install and set up Android Studio:
    - Android Studio Installation Guide

3.6

- Follow the guide to install and set up Android Studio:
    - Flutter Installation Guide

---

4. Open Project in Android Studio

1. Launch Android Studio:
    - Open Android Studio from your computer. Ensure you have the latest version installed for optimal compatibility with Flutter and any other dependencies.
2. Open Existing Project:
    - On the Android Studio welcome screen, select “Open an existing project.”
    - Select the project folder and click “OK” to open it.
3. Configure Project:
    - Android Studio will index and configure the project. It may prompt you to install missing dependencies or update project settings. Follow any on-screen instructions to complete the setup.
4. Verify Configuration:
    - Ensure that Android Studio recognizes your Flutter SDK and any required plugins. You can check this by going to File > Settings (or Android Studio > Preferences on macOS), and then navigating to Plugins and Languages & Frameworks > Flutter.

2. Run Flutter and Python Installation Commands

1. **Open Terminal in Android Studio:**
    - **Within Android Studio, open the Terminal by selecting View > Tool Windows > Terminal. This will open a terminal window at the root of your project.**
2. **Run Flutter Commands:**
    - **First, ensure Flutter is installed and properly configured by running flutter doctor in the terminal. This command checks for any dependencies and issues in your Flutter environment.**
    - **Next, use flutter pub get to fetch the Dart packages listed in your pubspec.yaml file. This ensures that all necessary libraries and dependencies are available for your project.**
    - **You might also want to use flutter clean to remove any build artifacts if you encounter issues, followed by flutter pub get to refresh your dependencies.**
3. **Run Python Installation Commands:**
    - **Activate the virtual environment. On Windows, use myenv\Scripts\activate, and on macOS/Linux, use source myenv/bin/activate.**
    - **Install the required Python packages by running pip install -r requirements.txt if you have a requirements.txt file, or use individual pip install commands for specific packages.**

3. Run the Frontend Folder

1. **Navigate to the Frontend Folder:**
    - **In the terminal, navigate to the directory where your frontend code is located. This could be a subfolder within your Flutter project or a separate directory.**
2. **Start the Flutter Application:**
    - **In the terminal, run flutter run to start your Flutter application. This command builds and runs the application on your connected device or emulator.**
    - **Ensure that your device or emulator is properly set up and running. You can check connected devices by using flutter devices.**
3. **Run Python Backend(run the audio server script)**

4. Have Fun with the App

1. **Explore Features:**
    - **Interact with the application by navigating through its features. Test different functionalities and ensure that everything works as expected.**
2. **Debug and Test:**

- **If you encounter any issues or bugs, use Android Studio's debugging tools to troubleshoot and resolve them. You can set breakpoints, inspect variables, and view logs to understand the behavior of your app.**

3. **Optimize and Enhance:**
   - **Consider any enhancements or optimizations you can make. This might include improving performance, adding new features, or refining the user interface.**
4. **Share and Collaborate:**
   - **If you're working with a team, share your progress and gather feedback. Collaborate to address any issues or incorporate new ideas.**

Enjoy working with your app! If you need any specific advice or run into problems, feel free to ask.

By following these steps, you will set up all the necessary software and dependencies for the Moodmix project. If you encounter any issues, refer to the provided installation guides or seek additional support online.
