PREAPARED BY :

DevDynasty

MoodSphere

Installation Manual

Installation Manual

INDEX

Project Overview:

MoodSphere is an unique online application that aims to change the way people engage with music and language learning. MoodSphere uses powerful picture recognition algorithms to estimate users' current emotional states based on photographs uploaded or shot in real time. This data is then utilized to design music recommendations that not only suit the current mood but also improve language learning experiences.

Introduction:

This manual provides comprehensive guidance on deploying both the frontend and backend components of the MoodSphere application, an emotion-based music recommender system. In this document, we cover system requirements, software requirements, installation procedures, and the deployment process for the application. Detailed steps are provided to ensure even those new to such deployments can successfully deploy and maintain the application.

Overview of Subtopics:

MoodSphere is an unique Web application that aims to change the way people engage with music and language learning. MoodSphere uses powerful picture recognition algorithms to estimate users' current emotional states based on photographs uploaded or shot in real time. This data is then utilised to design music recommendations that not only suit the current mood but also improve language learning experiences.

The major factors covered to deploy the project locally are in this manual as follows:

1. ***Checking up the System Requirements***: This section outlines the hardware and software prerequisites for running the MoodSphere application efficiently

2. **Software Requirements and Installation Guides:** Detailed instructions for installing all necessary software, including Git, Node.js, npm as well as setting up your development environment on both Windows and macOS systems.

3. **Error Handling and troubleshooting:** there might be few cases where you could face some errors in frontend as well as in backend to the application.

Need for System

As our product doesn't have any specific needs but due to its integrity with the algorithm we will require few basics components which has listed as below:

A. Checking up the System Requirement

Hardware System Requirements:

Processor (CPU): Core i5/i7-U 1.5 Ghz or AMD equivalent or For Mac OS: Apple M1 (8-core)

Memory (RAM): 8 Gb

Available Storage on Disk: Minimum: 10 Gb of SSD , Maximum: 20 Gb of SSD or more.

GPU: Intel Iris Xe with 1.30 Ghz

Software System Requirements:

- **Operating System**: Windows 10/11 or macOS Big Sur or newer

Softwares Installations

Software Used :

- **Git**
- **Visual Studio Code (recommended IDE)**
- **Node.js**
- **npm (Node Package Manager)**

How to install this softwares? Guidelines:

Before starting the deployment process, ensure you have the necessary tools installed on your system. Below are the guides for installing Git, Visual Studio Code, Node.js, npm, and creating a Vercel account on Windows and macOS.

Git:

- **Windows**: Download and install from Git for Windows[https://gitforwindows.org/]
- **macOS**: Install via Homebrew with brew install git, or download from Git SCM [https://git-scm.com/download/mac]

Visual Studio Code:

- **Windows & macOS**: Download and install from Visual Studio Code [https://code.visualstudio.com/Download]

Node.js and npm:

- **Windows & macOS**: Download and install from Node.js official site. npm is included with Node.js. [https://nodejs.org/en]

Once you have all the software list installed as mentioned above you can now follow through further steps to deploy the project on your system locally.

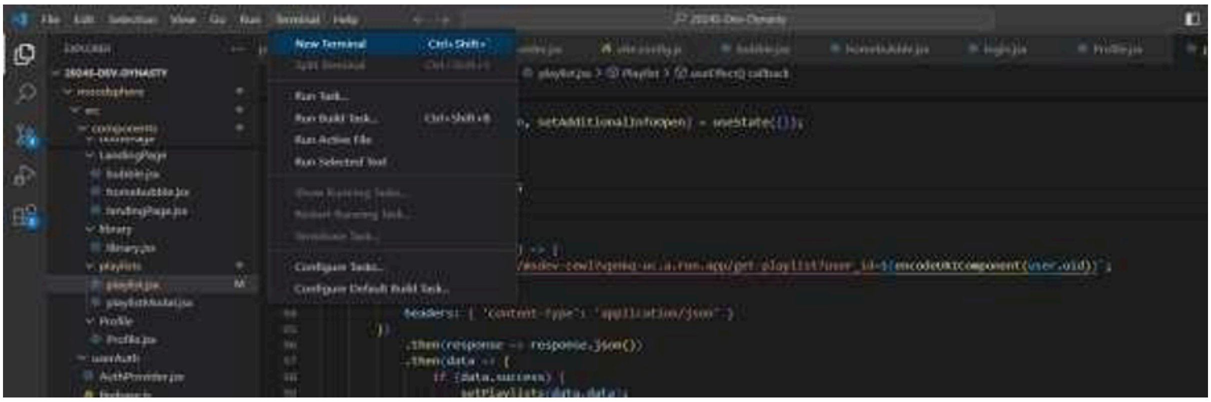
Setting up FRONT-END & BACK-END

Step 1: Clone the Repository

- Open your web browser and go to the GitHub repository at https://github.com/htmw/2024S-Dev-Dynast
- Locate the green "Code" button on the GitHub page and click it.
- Copy the URL under "Clone with HTTPS".
- Open a terminal on your computer.
- Choose or create a directory on your local machine where you want to clone the repository.
- Type git clone, then paste the URL you copied. It should look something like this:
  - git clone https://github.com/htmw/2024S-Dev-Dynasty.git
- Press Enter. Your local clone will be created.

Step 2: Set Up the Project

- Once the cloning is complete, navigate to the cloned directory by typing:
- “ cd 2024S-Dev-Dynasty ”
- Open new terminal.

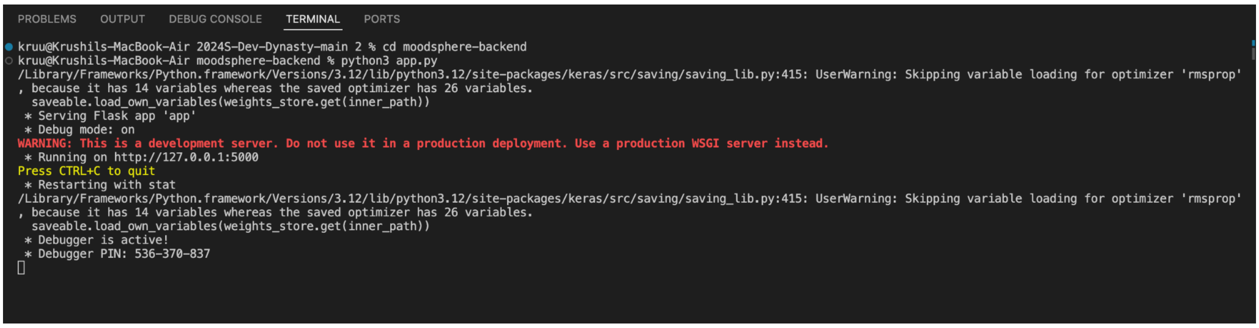


- If your project has multiple subdirectories and the one you need to deploy is 'moodsphere', enter:
- “ cd moodsphere ”
- Onto which run “ npm i ” to check for additional npm packages.
- Once you have reached into moodsphere folder, run “ npm run dev ”

This would set you up with the front-end deployment locally.\

Step 3: Run the Application Locally for backend:

- **Open the new terminal.**
- Enter the command " **cd moodsphere-backend** "
- **After into the backend folder run the command: " python/python3 app.py "**
- This will lead to development of backend locally in VSC.



```
● kruu@Krushils-MacBook-Air 2024S-Dev-Dynasty-main 2 % cd moodsphere-backend
○ kruu@Krushils-MacBook-Air moodsphere-backend % python3 app.py
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages/keras/src/saving/saving_lib.py:415: UserWarning: Skipping variable loading for optimizer 'rmsprop'
, because it has 14 variables whereas the saved optimizer has 26 variables.
  saveable.load_own_variables(weights_store.get(inner_path))
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages/keras/src/saving/saving_lib.py:415: UserWarning: Skipping variable loading for optimizer 'rmsprop'
, because it has 14 variables whereas the saved optimizer has 26 variables.
  saveable.load_own_variables(weights_store.get(inner_path))
 * Debugger is active!
 * Debugger PIN: 536-370-837
```

As you will be able to see the output as above you have successfully installed all the libraries.

Here's an additional step if required to set up your virtual environment

Step One:

- Install **Python.**
- Ensure that Python is installed on your machine. You may get it from the official Python website. Ensure that you have Python 3.x, which contains the **venv** module.

Step Two:

- Create a Virtual Environment & Open the terminal or command prompt.
- Navigate to the project directory where you wish to configure the virtual environment.
- To create the virtual environment, use the following command:
  " **python -m venv myenv** "

Step Three:

- Activate the Virtual environment.
- On windows you can use "**python -m venv myenv** ".
- On MacOS you can use " s**ource myenv/bin/activate** ".

Step Four:

- Install the packages by using " **pip install Flask keras Pillow pandas Flask-Cors firebase-admin numpy** " if you want to exit the virtual environment then you just have to run " **deactivate** " and it will end.

As walking through this installation manual We would like to suggest that if you are trying to access the firebase-admin you need to reach out to us for a Private key. Kindly email us at dynastydev23916@gmail.com.

Common Errors & Troubleshooting

While installing the Mood Sphere there might be few errors which we might think can be common to each user’s out there. We have solved a few common errors as listed.

- **Npm not found / axios error :**
- This can be because of installing the NPM.
- Open the new terminal and try to install the NPM as by downloading “ https://nodejs.org/en/download”
- Check for its version by entering - “ node -v”
- If the problem persists, try the command “ npm i ” which will resolve the problem based on the node module.

- **Python not found:**
- This error usually is because of the inappropriate installation for the Python.
- Delete the python and its launcher from your system and download it again. “ https://www.python.org/downloads/ ”
- Once you download, Kindly reverify that your system path and your file location have to be in same directory.
- If you can not find the path, you can always check as Settings > Environment Variable > set a new path as administrator > select the path which have same directory > Apply and exit.
- This can be reverified in the CMD (Command Prompt) using the command “ python -v” to check the version.

And you are all set to GO and Run it again.!

Reach out to us

We value your feedback and are here to assist you with any inquiries or concerns you may have regarding the MoodSphere application deployment process. Feel free to reach out to us via email at ***dynastydev23915@gmail.com****. Whether you have questions about the deployment steps, encountered any issues during the process, or have suggestions for improvement, we welcome your input.*

Additionally, if you have any feedback or ideas for enhancing the MoodSphere application, we would love to hear from you. Your insights play a crucial role in shaping the future development and functionality of our application. We are committed to delivering an exceptional user experience and greatly appreciate your input.

Thank you for choosing ***MoodSphere****. We look forward to hearing from you! Warm regards,*

From Your one's and only - Team **DevDynasty**