Retail Granite Wiki Chatbot Setup Guide

This document provides step-by-step instructions on how to set up the Retail Granite Wiki Chatbot project from the GitHub repository. Follow the steps below to install and run both the front-end and back-end of the application.

# Clone the Repository

To get started with the project, clone the Git repository from the following URL:  
  
Git Repository URL: [https://github.com/jit4ibm/ChatBot.git](https://github.com/jit4ibm/ChatBot.git)  
  
To clone the repository, use the following command in your terminal:  
  
git clone https://github.com/jit4ibm/ChatBot.git  
cd ChatBot

# Install Dependencies

Navigate to the project folder and install all the required dependencies:  
  
 npm install

npm install web-vitals

# Run the Backend Server

Start the backend server:

node src/Retail\_bee\_wiki\_granite\_server.js  
This will run the Express server, which will be accessible on port 4000 by default.

# Start the Frontend Development Server

To start the React development server for the frontend, run:  
  
 npm start  
This will launch the React app, typically accessible on http://localhost:3000.

# Verify the Communication

Ensure that the front-end React app is correctly communicating with the back-end Express server. You can test the interaction by opening the front-end in your browser and checking if the API calls are being correctly made to the back-end.

# Check the Backend Port

Ensure the back-end is running on an appropriate port, like 4000. If necessary, adjust the port number in the backend code (e.g., in Retail\_bee\_wiki\_granite\_server.js). The default backend port is 4000.

# Environment Setup (Optional)

If the environment requires additional configurations (such as API keys, environment variables, etc.), ensure to create a .env file in the root directory. Example .env file:  
  
  
REACT\_APP\_API\_URL=http://localhost:4000  
OLLAMA\_API\_KEY=your\_api\_key  
 These can be accessed in your React app using process.env.REACT\_APP\_API\_URL.

# Test the Frontend and Backend

After setting up both the front-end and back-end servers, test if both are working. Open the React app and ensure it is making API calls to the back-end server.

# Production Build (Optional)

Once everything works, you can create a production build of the React app by running:  
  
npm run build  
This will create a static build of your front-end code, which can be served through your back-end or a different static file server.

# Troubleshooting

If you encounter any errors during the setup, ensure all dependencies are installed correctly. Check the back-end server's port and ensure no port conflicts with the front-end. If something is missing, it will likely appear as an error during npm install. To resolve missing packages, run:  
  
npm install <missing-package-name>