**Setting up and Running the Project Documentation**

This documentation provides step-by-step instructions on how to set up and run the project. Follow these steps to get the project up and running on your local development environment.

**Prerequisites:**

* **Python:** Ensure that Python is installed on your system. You can download and install Python from the official website (<https://www.python.org>).
* **Django**: Install Django using the following command: “ pip install Django ”.
* **PostgreSQL**: Install PostgreSQL database on your system. You can download and install it from the official website (<https://www.postgresql.org>).
* **Some warnings regarding missing libraries/modules might pop up in the terminal while trying to run the project. That’s nothing much to be worried about. Simply read the name of missing module and run the ‘pip’ command to install the missing module.**

**Steps:**

**1. Clone the Repository:**

- Open your terminal or command prompt.

- Navigate to the directory where you want to clone the project.

- Run the following command to clone the repository:

```

**git clone <repository-url>**

```

**2. Create a Virtual Environment:**

- Navigate to the project directory.

- Create a new virtual environment by running the following command:

```

**python -m venv myenv**

```

- Activate the virtual environment:

- On Windows:

```

**myenv\Scripts\activate**

```

**- On macOS and Linux:**

```

source myenv/bin/activate

```

**3. Install Dependencies:**

- Ensure that the virtual environment is activated.

- Run the following command to install the project dependencies:

```

pip install -r requirements.txt

```

**4. Set Up the Database:**

- Ensure that PostgreSQL is installed and running on your system.

- Open the project's settings.py file located in the project's directory.

- In the DATABASES section, update the database settings with your PostgreSQL credentials (e.g., database name, username, password).

**5. Migrate the Database:**

- Run the following command to apply database migrations:

```

python manage.py migrate

```

**6. Create a Superuser:**

- Run the following command to create a superuser for accessing the Django admin panel:

```

python manage.py createsuperuser

```

- Follow the prompts to enter a username, email (optional), and password.

**7. Run the Development Server:**

- Start the development server by running the following command:

```

python manage.py runserver

```

- The server will start running at http://localhost:8000/.

**8. Accessing the Admin Panel:**

- Open your web browser and go to http://localhost:8000/admin/.

- Log in using the superuser credentials created earlier.

**9. Testing the API Endpoints:**

- Use a tool like Postman to test the API endpoints.

- Refer to the API documentation or the project's views.py file to understand the available endpoints and their functionalities.

**Congratulations!** You have successfully set up and run the project on your local development environment. You can now start exploring and interacting with the REST API.