

Setting Up the Backend:

1. Install Node.js and PostgreSQL:

- Make sure Node.js and PostgreSQL are installed on the server. You can follow the official documentation for installation instructions:

- Node.js Installation Guide: <https://nodejs.org/en/download/>
- PostgreSQL Installation Guide: <https://www.postgresql.org/download/>

2. Copy the Backend Folder:

- Upload the `backend` folder to the desired location on the server.

3. Environment Variables:

- Navigate to the `backend` folder.
- Locate the `.env` file.
- Edit the following variables:
 - `ADMIN_PASSCODE`: Change it to the desired passcode.
 - `DATABASE_HOST`, `DATABASE_USERNAME`, `DATABASE_NAME`, `DATABASE_PASSWORD`: Update these variables according to your PostgreSQL setup.

4. Install Dependencies:

- Run `npm install` inside the `backend` folder to install all the required dependencies.

5. Database Setup:

- Ensure that PostgreSQL is running.

6. Start the Server:

- Run `npm run start` to start the backend server.

Setting Up the Dashboard:

1. Copy the Dashboard Folder:

- Upload the `dashboard` folder to the desired location on the server.

2. Environment Variable:

- Navigate to the `dashboard` folder.
- Locate the `.env` file.
- Edit the `REACT_APP_API_URL` variable to point to the backend API URL on the server.

3. Build the React App:

- Run `npm install` to install all the required dependencies.
- Run `npm run build` to build the React app for production.

4. Serve the Dashboard:

- Use a web server like Nginx or serve the static files directly with Node.js to host the built React app.

5. Configuring Nginx (Optional):

- If you choose to use Nginx, you'll need to configure it to serve the built React app.

6. Restart Nginx:

- After configuring Nginx, restart the Nginx service to apply the changes.