Trym Todo App

Structure:

Basic task tracking application built using a Django backend with a ReST framework, a MySQL database hosted in Amazon RDS, and a React frontend. Testing is run through pytest for the backend, and a combination of mocha, chai, enzyme, and sinon for the React frontend. I also used Postman to test my Django api, which is available in the github through the readme.

The frontend utilizes prettier, eslint, and reactstrap packages and was built in VS code.

Instructions:

To run the application locally (Windows) open Windows Powershell and navigate to the project root.

1. Create a .env file and paste in the contents at the bottom listed under .env file. Place the file in the backend directory, like in the screenshot shown at the bottom.
2. Open a virtual environment using the pipenv command: pipenv shell
3. Next cd into the backend folder and run the server with: python manage.py runserver
4. Using a new terminal (I use git bash) navigate to the client folder inside the client folder and start the frontend using: npm start
5. Testing can be run using the command: “pytest” in the root directory in a powershell terminal.
6. To test the react frontend navigate to the client folder and run command: npm test
7. To run the postman test, install postman, and then click the link in the github readme. It will show tests for all CRUD endpoints and pass/fail tests for each.
8. The Django backend is written using Django admin, so the API can be interacted with through that framework. With the server running navigate to: <http://localhost:8000/admin> and sign in with the credential at the bottom of this page.
9. Backend documentation can be found at: <http://localhost:8000/admin>/docs

**Django admin credentials**

Admin

[Jordanvaughn1@gmail.com](mailto:Jordanvaughn1@gmail.com)

Trymtodoapp1!

Contents of .env file:

USER=user

PASSWORD=MeGGHH8X48PcQXL

HOST=tododb.cog9jhfqmf6z.us-east-1.rds.amazonaws.com

SECRET\_KEY=5of^5ign%&(+!$ddkw%l^d80i6s(3s5d3lxrmvx7)20%afn#7%

![A screenshot of text

Description automatically generated]()