A full-stack website with React and Django deployed using AWS.

This repository https://github.com/akibjawad/django-react contains the source code of a website I developed for an interview stage. This website helps user in showcasing his/her/their completed projects along with the source code (git url). It is a multi-user web app where each user can maintain their separte list of projects. Like a standard full-stack website, it has 2 completely isolated modules: 1. Frontend module is built using `React` with `vite` bundler 2. Backend module is built based on `django-rest-framework`

Frontend with (React + Vite)

The frontend website basically provides UI to perform these three functions: 1. Register and Login with username and password

2. After completing authentication (register and login) user can add their project details (title, description, and Github URL of the source code) in the portal. Providing GitHub URL is optional, If user does not provide, web app will keep an empty link with `#`.

3. Users will also be able to view and delete their previously added projects

While developping, I tried to keep most commonly used elements and concepts of React such as Navbar, Forms, Modal with Overlay, and Protected Routing etc.

Backend with Django REST Frameworks and SQLite DB

Backend of this website supports the frontend with 2 `API` endpoints: 1. First API endpoint (`api/user/`) is used to create and authnticate users

2. Second API endpoint (`api/projects/`) is used for creating and deleting projects.

Just like the Frontend, I tried to keep most commonly used practice of Django development such as creating Serializers, Models, and Views. I tested out multiple database engines, but eventually I settled for the default `sqlite` db to keep the simplicity of this project.

Run the project locally

Two run locally, from the project root we need to go the backend directory and run the `run-local.sh` script. Commands are provided below.

```
cd backend
chmod +x ./run-local.sh
./run-local.sh
```

1

After starting backend, user also need to run the frontend. To run the front end user simply need to run the `run.sh` script in the front end directory. Commands to run the frontend from the project root directory is provided below.

```
chmod +x ./run.sh
cd frontend
./run.sh
```

Deploying it on AWS

I deployed the backend (Django server) on an AWS EC2 while the frontend (React App) on an AWS S3 site. I used both of AWS service as a FREE-Tier service. Whole website can be found at http://ajreactdemo.s3-website-us-west-2.amazonaws.com

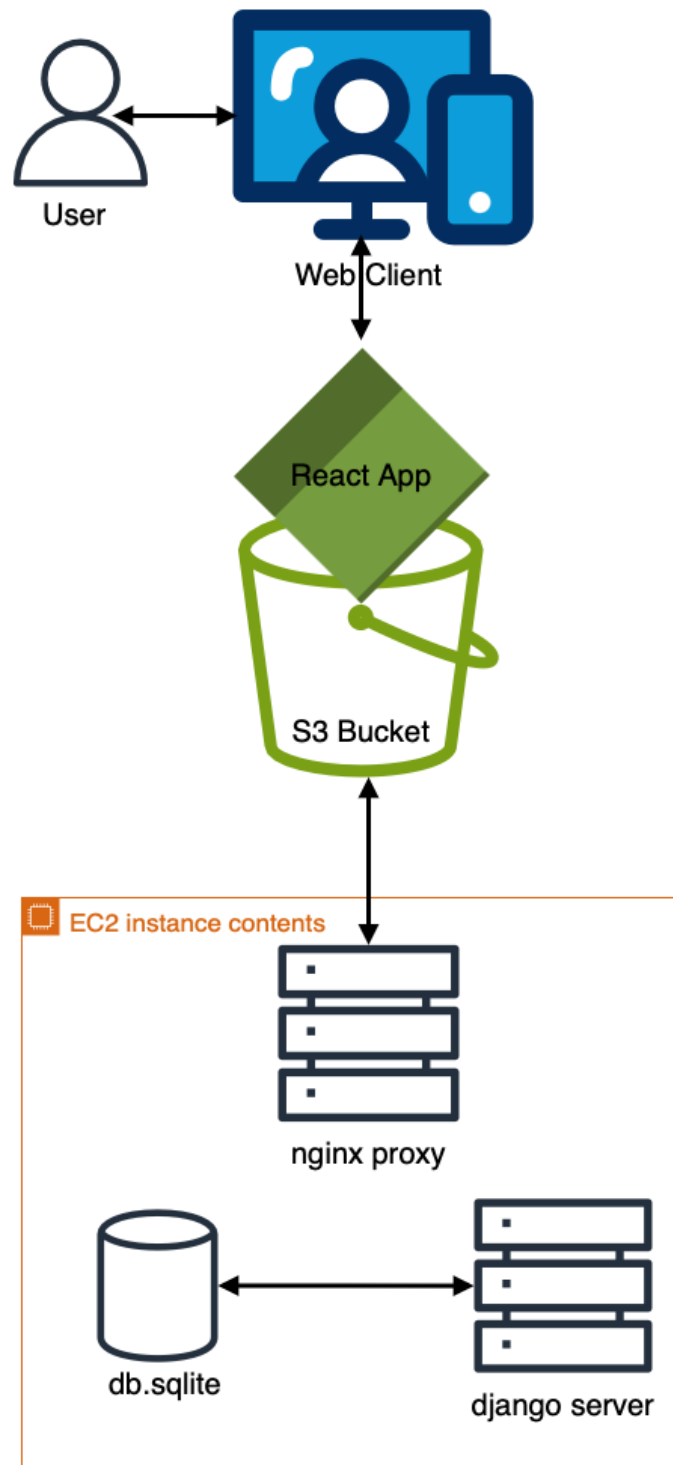My deployment architecture is also provided in the following image for better clarrification.

Figure 1: Deployment Architecture