

Website Operation Presentation

Working with React and Django for this project was a first for me. Previously, I had worked with React in a local environment and did not have to store data in a database, but with this project, I got to expand on that front and create a full-stack website that incorporates both a front end and a back end.

As mentioned above, for the front end of this project I used React + Vite. React is a potent tool that allows you to separate your project into components, and use them separately as building blocks to build your building/project.

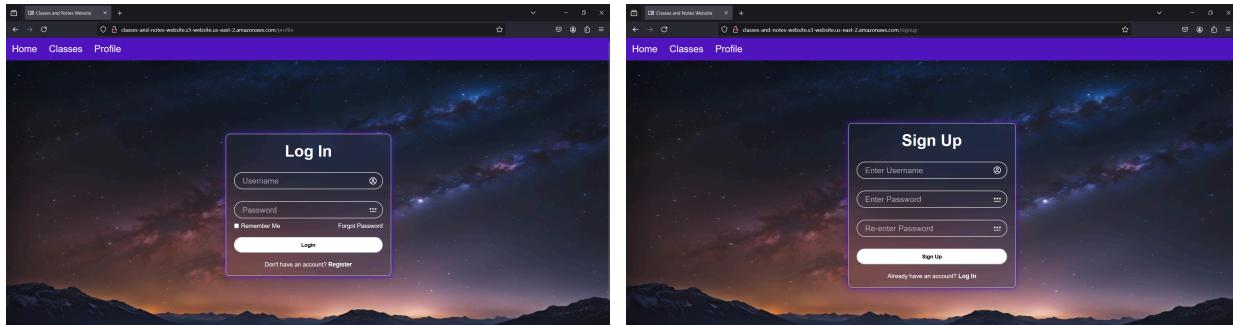
For the back end, this project uses Django, a high-level framework that is used to build fast and efficient websites using Python. Django allows developers to build objects through endpoints, removing the need for the developer to build the entire back end from scratch. This allows fast scalability and efficiency in the work environment. Below is an image of the endpoints used in this project:

```
backend > backend > urls.py > ...
1  from django.contrib import admin
2  from django.urls import path, include
3  from api.views import CreateUserView
4  from rest_framework_simplejwt.views import TokenObt
5
6
7  urlpatterns = [
8      path('admin/', admin.site.urls),
9      path('api/user/register', CreateUserView.as_vie
10     path('api/token', TokenObtainPairView.as_view()
11     path('api/token/refresh/', TokenRefreshView.as_
12     path('api-auth/', include("rest_framework.urls"
13     path('api/', include("api.urls")),
14 ]
15
```

```
backend > api > urls.py > ...
1  from django.urls import path
2  from . import views
3
4  urlpatterns = [
5      path("notes/", views.NoteListCreate.as_view()),
6      path("notes/delete/<int:pk>/", views.NoteDelete
7      path("classes/", views.ClassListCreate.as_view(
8      path("classes/delete/<int:pk>/", views.ClassDel
9 ]
```

These endpoints are used for creating new users, generating a log-in and refresh token, and creating notes and classes on the website. They allow the front end to make API calls to the server, and the server takes care of the rest and gives a concise response to the front end with the required information.

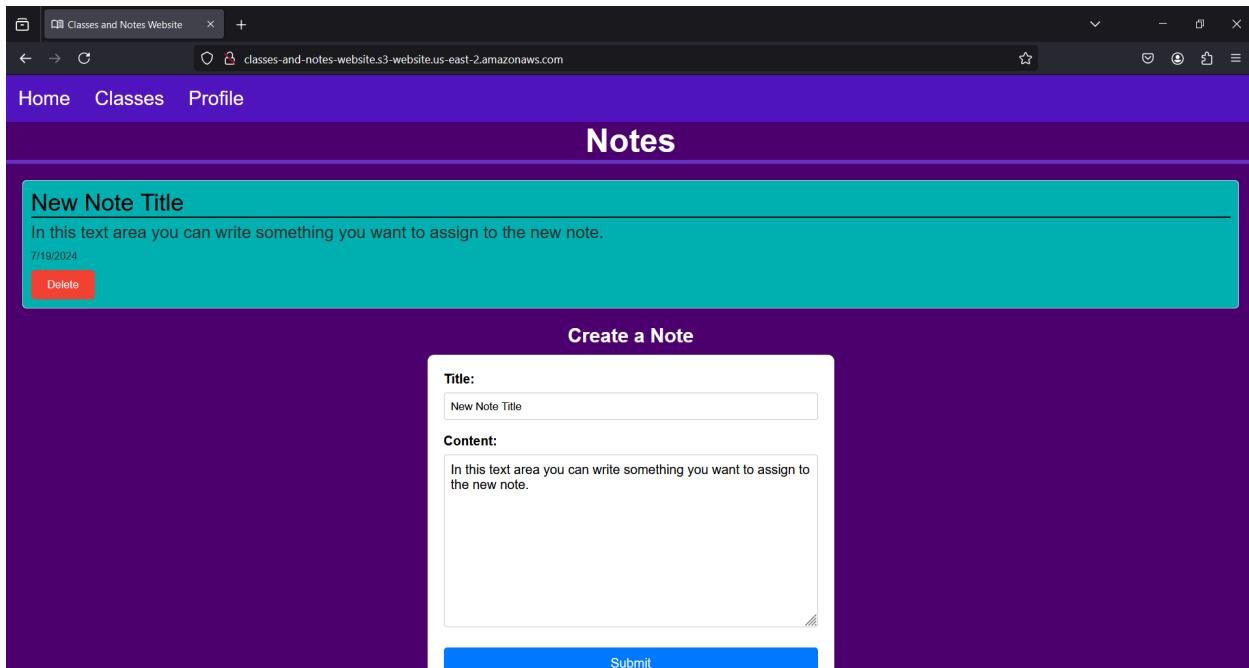
Now, the website itself is a simple website that is meant to display the usage of the back end and the API calls. It requires the user to log in and register, creating an account. It is important to note that the “Remember me” and “Forgot Password” on the login page currently do not work.



Log In page

Sign Up Page

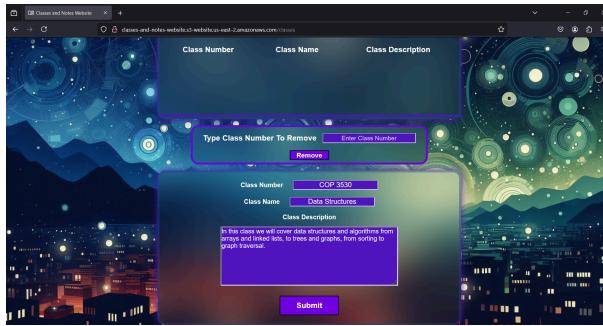
Once you have logged in, you will be able to use the Home and Classes pages. The Home page will allow you to write notes and save them in the database to retrieve them in the future. You can add new notes and also delete already existing ones.



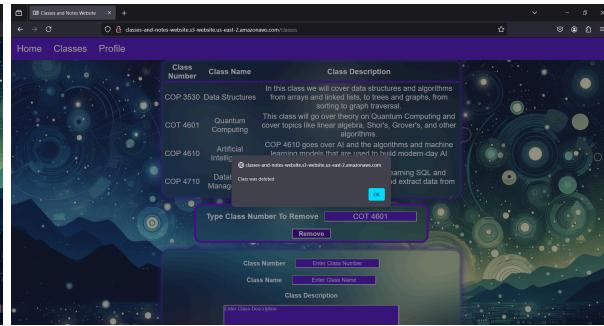
The home page where you can add notes

The classes page will allow you to create a list/table of all the classes you take part in. You can add new classes using the bottom entry option and remove an entry using the middle component where you can enter the class number and remove an

existing entry in the table. Both these features will make changes to the database through the Django server.



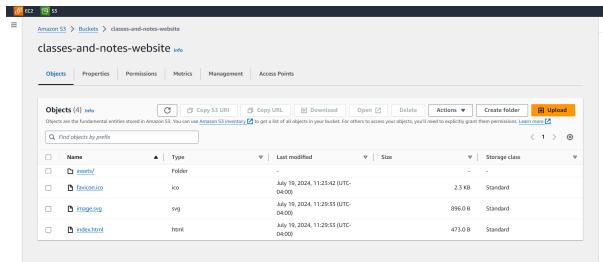
Adding a new class entry



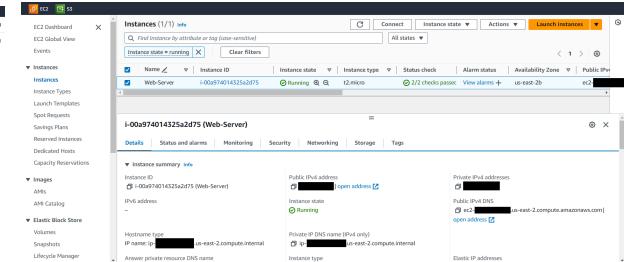
Removing an entry from the table

The final page is the profile page. Currently, it does not display much, but it allows you to log out of the website and bring you to the landing page where you have to re-enter your credentials to log in and access all the information you created, or you can create a new user through the Register button at the bottom.

The website is deployed using the AWS platform. For the front end, I am deploying it on an S3 bucket, which provides a simple layout and deployment of the website, and for the back end, I am using an EC2 instance, which allows you to access a CLI where you can make scripts and let the server running non-stop.



AWS S3 Bucket Running the Front End



AWS EC2 Instance Operating the Back End

Below are all the relevant links where you can view the source code and the website itself.

Website link - <http://classes-and-notes-website.s3-website.us-east-2.amazonaws.com>

Front End Github Page - <https://github.com/INIT27/react-website>

Back End Github Page - <https://github.com/INIT27/Website-Backend>