CNIT 58100-048 Milestone 3

Summary

YourTube provides users the possibility to maintain separate profiles while consuming video content. Milestone 3 consist of UI upgrade to the application using React.js framework. The backend is setup using Django and 2 models have been created for YourTube users and videos. The frontend renders the list of videos and managers users from the SQLite database.

The intended users are all the users of YouTube who wish to have sense of control over their watch time and type of content according to their motivation.

Users will be able to toggle between "Entertainment" and "Information" profile. They will also be able to disable or enable the recommendations. All the fields in the table and views are minimalized to accommodate the requirements of this milestone. Communication to user (success or error) is communicated to the user using Alerts.

Application Setup

We will have to setup the frontend and backend before accessing the YourTube application.

Kindly download the folder from GitHub.

Backend- Django Setup

> - cd YourTube_Backend_Django
> - pipenv shell
> - pipenv install
> - cd yourtube
> - python manage.py runserver

We are good to go. The backend is up and running. Now get back to the main folder.

Frontend – React Setup (Make sure you have Node.js and npm installed in your local machine)

> - cd YourTube_Frontend_React
> - npm install
> - npm start

We are all set to explore the application. Go to a browser and type http://localhost:3000.

This will direct you to http://localhost:3000/CNIT581-048-Milestone3.

Details

User Account

I have created a default user for you to login and explore. The credentials are as follows,

Email: **cnituser@purdue.edu**
Password: **Milestone3***

The app has a register page which will onboard new users. Feel free to create a new user credentials for yourself if you want to. The credentials will be stored to your local SQLite DB.
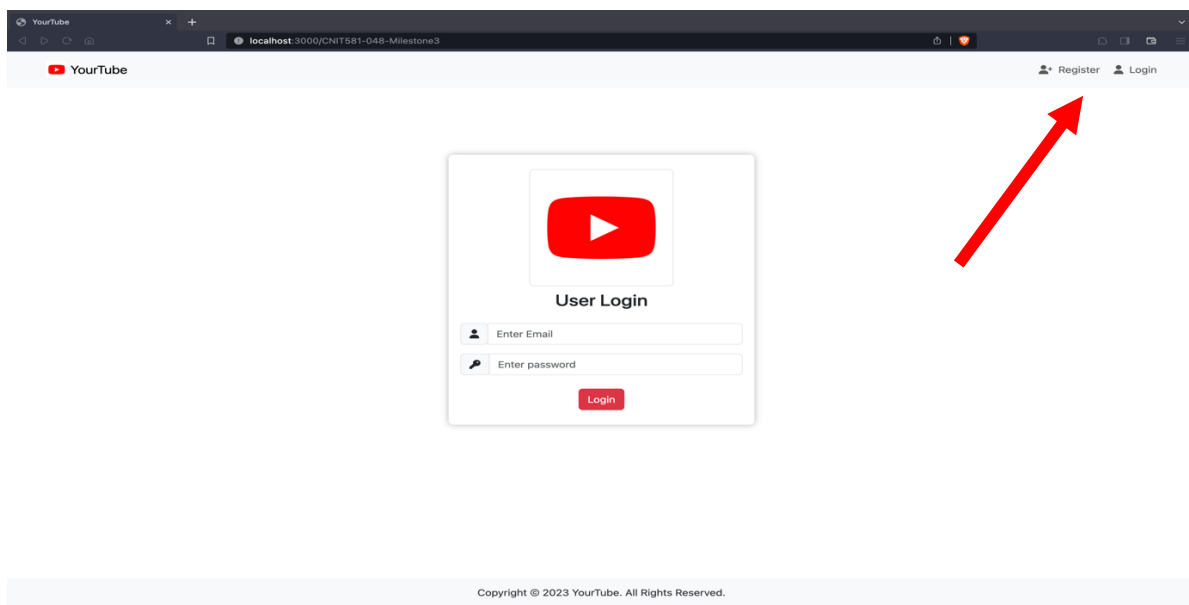
URLs

Go to a browser and type http://localhost:3000.

This will direct you to http://localhost:3000/CNIT581-048-Milestone3.

You will be able to access only the /login, /register and /CNIT581-048-Milestone3 routes when you are not logged in.

Views and Templates

Home View

User will be taken to this landing screen. Register and Login are the actions allowed before getting into the application. You must be logged in to explore the other views of the application.

List View
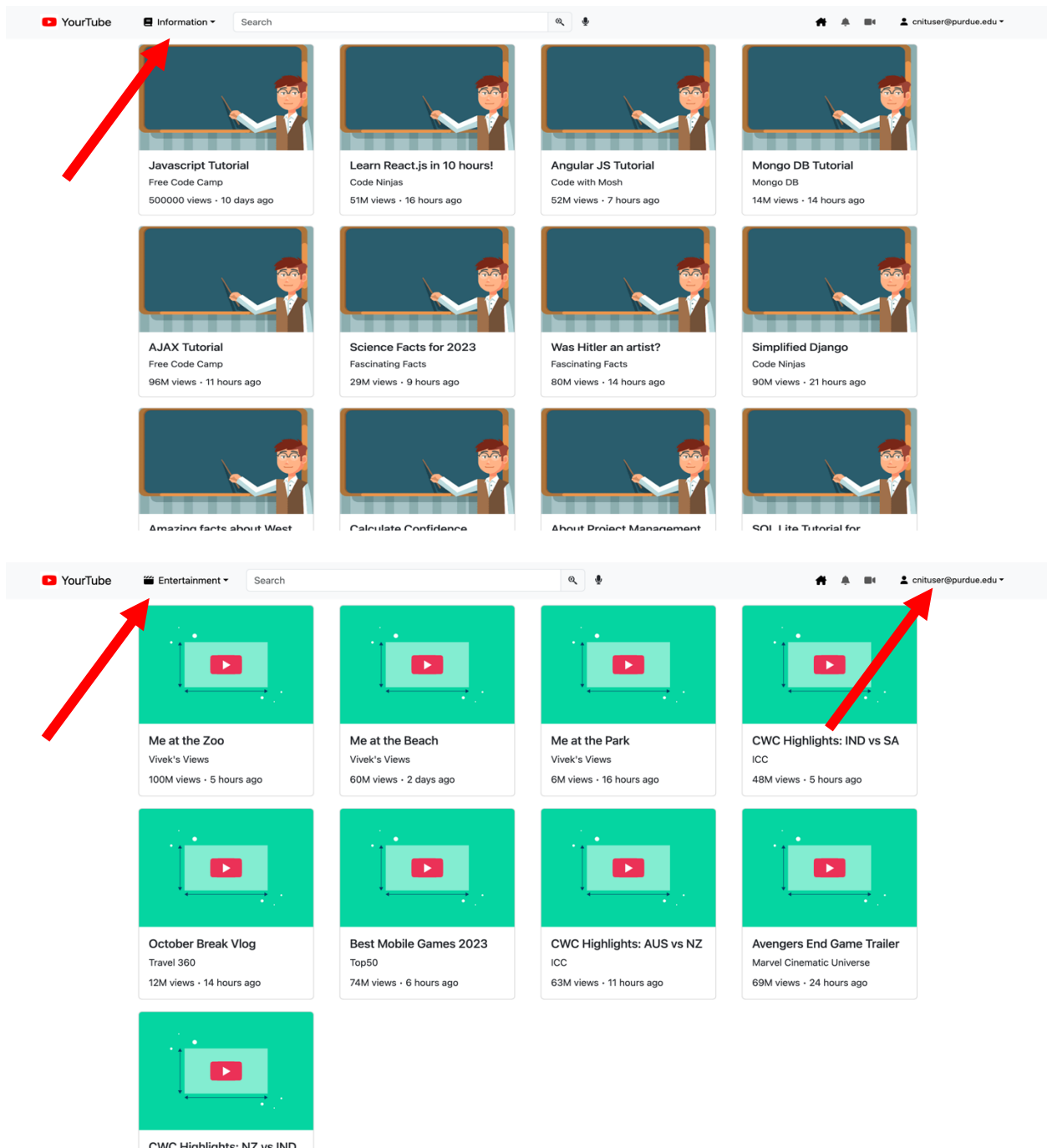
The Homepage lists the list of videos based on the user profile. User will be able to toggle between the profiles using the dropdown given near the search box. The list of videos will change dynamically based on the user input which is the video category of "Entertainment" and "Information".

The Navbar will have the user email ID indicating that the user has logged in to the application.

Detail View

For detail view, user can see their own profile by clicking the dropdown denoted by the user email and choose "My Profile" from the navigation bar. This will take the user to a detailed page (/profile) which displays the information of the user. This is retrieved using the 'GET' method. Backend authenticates and checks the authorization the API call before sending out the response.
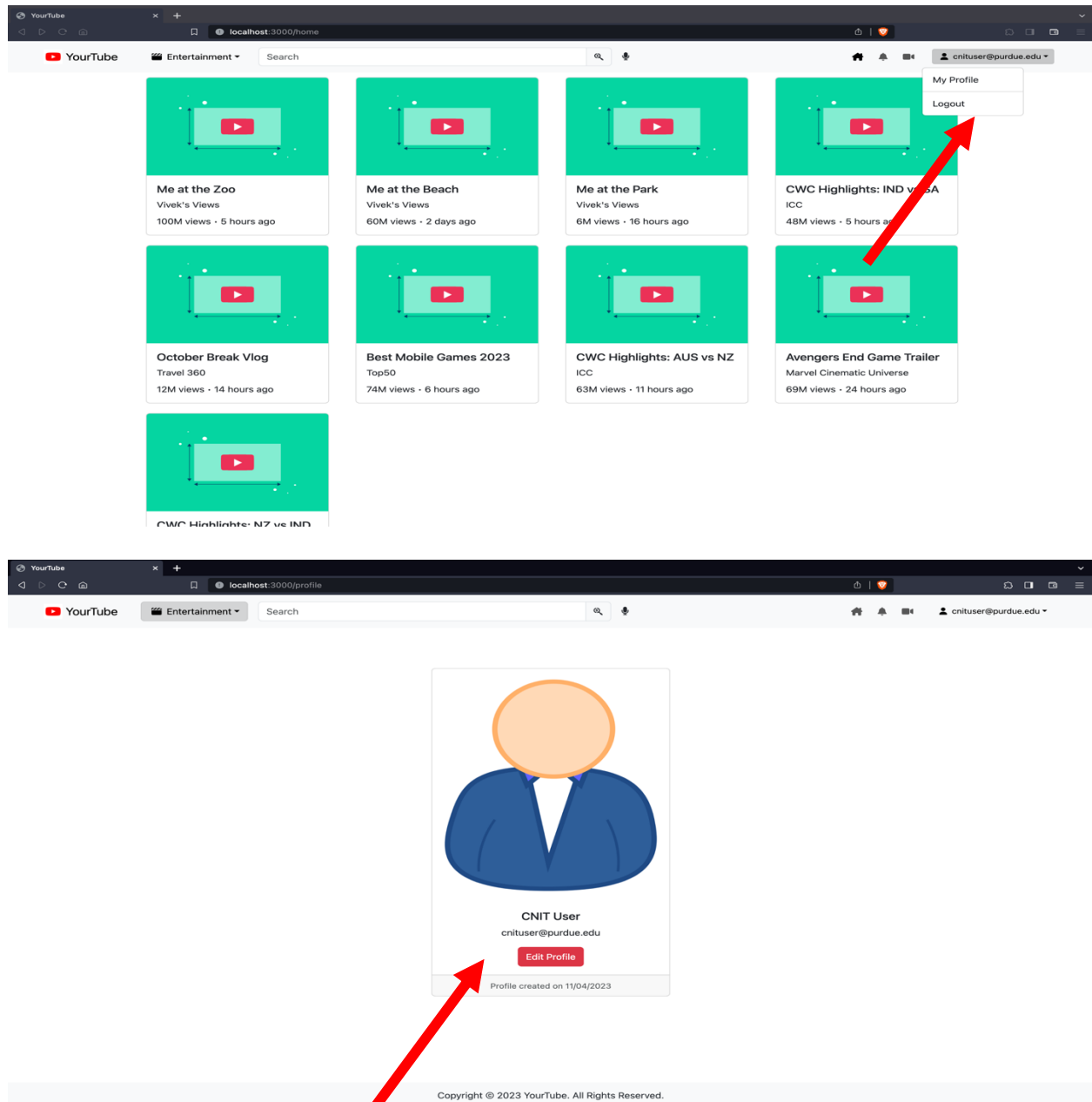

This page displays the user's name, email address and the date in which the profile was created on along with the user profile image (yet to be integrated, as of now the image is static). The page has a "Edit Profile" button which will allow us to update the user details.
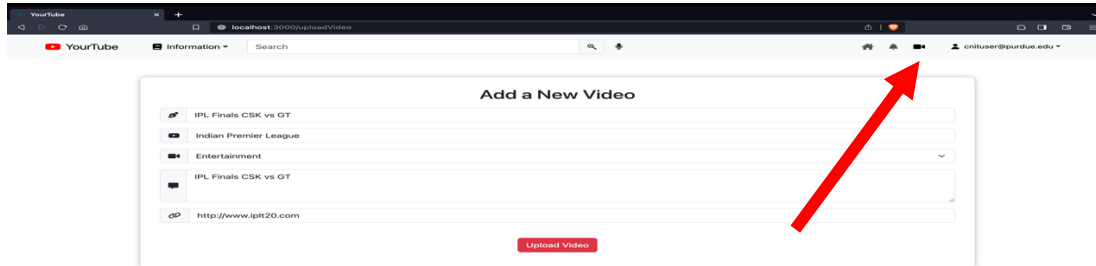
Edit View
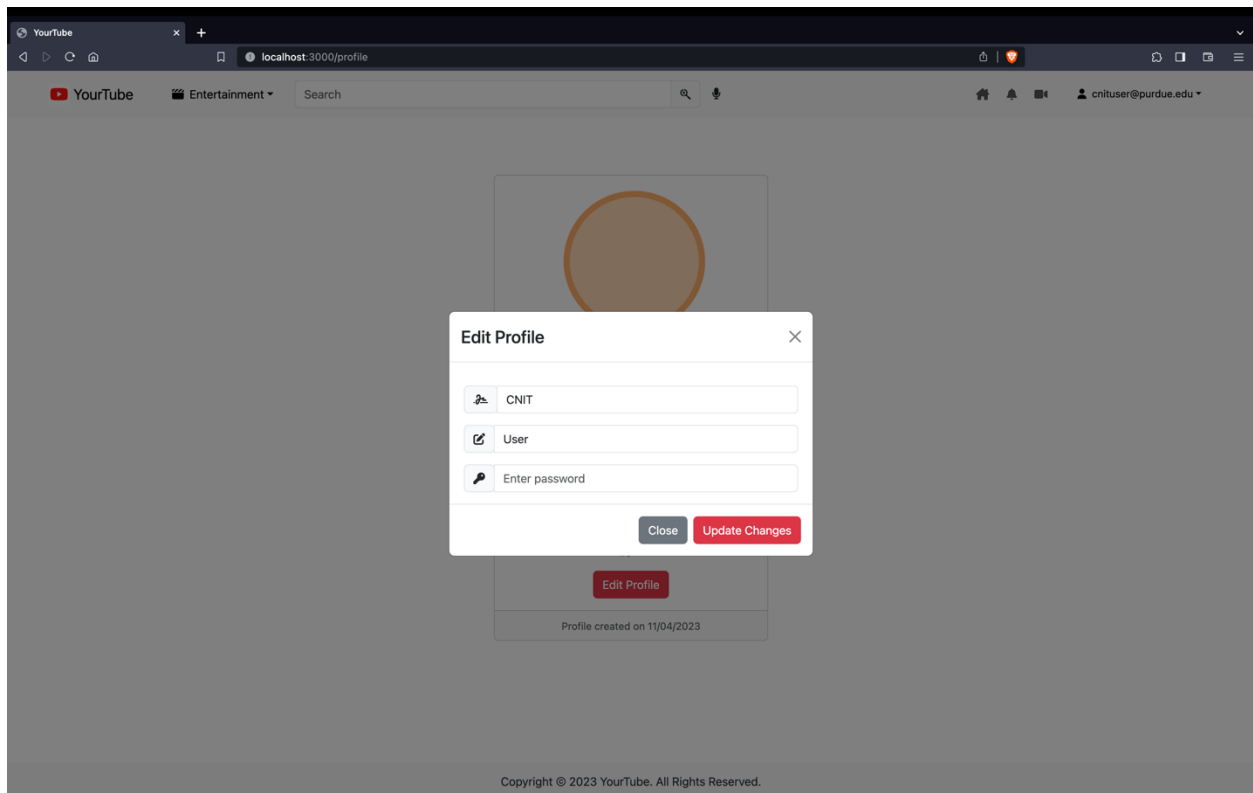
This view will allow us to update the existing information of the user including first name, last name and password (more fields to be added). This uses 'PUT' method and needs authentication.
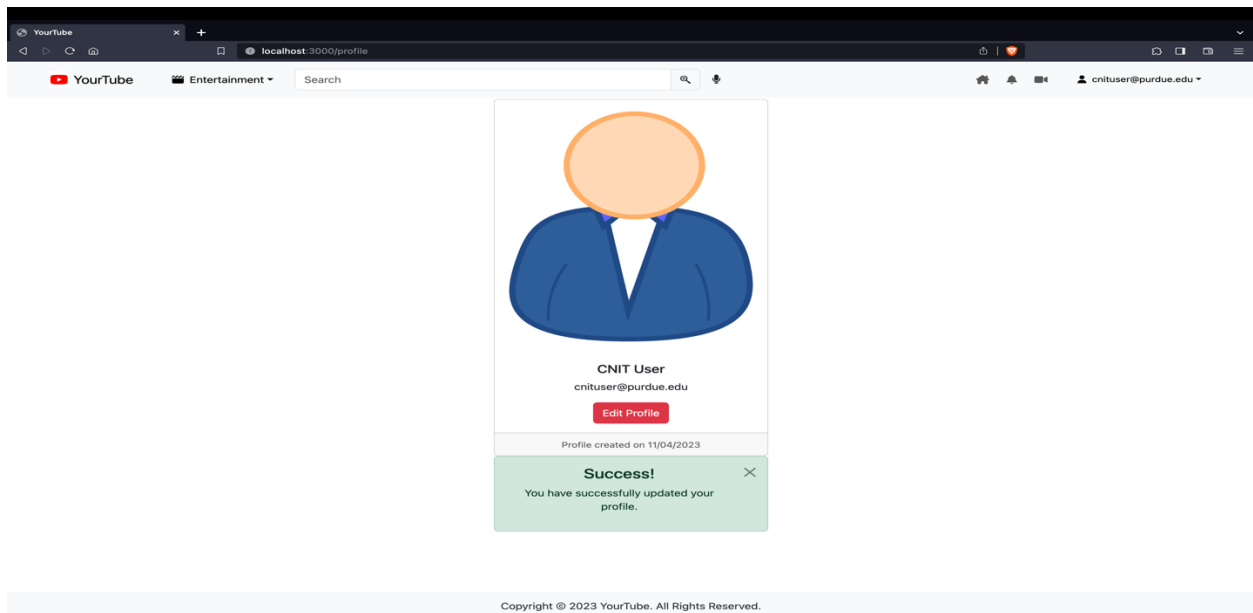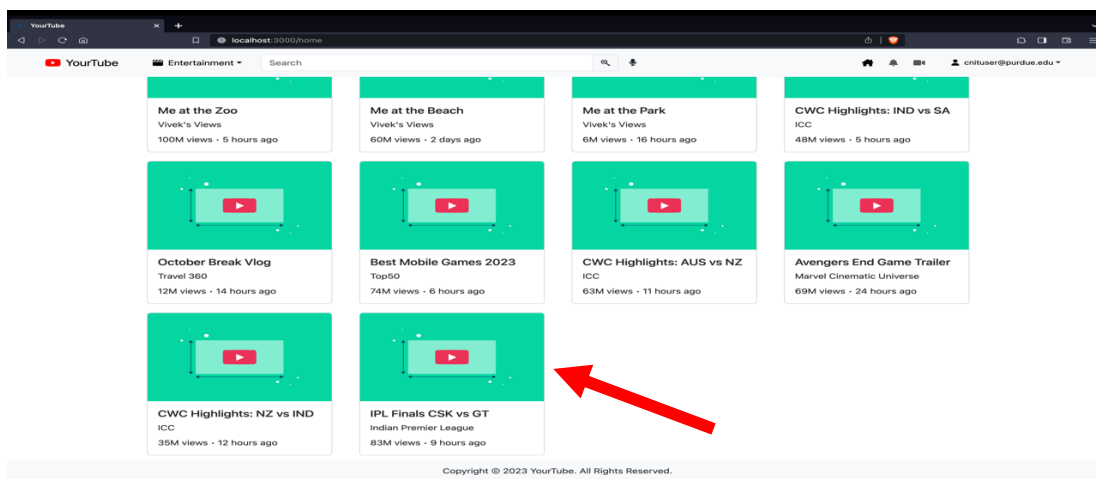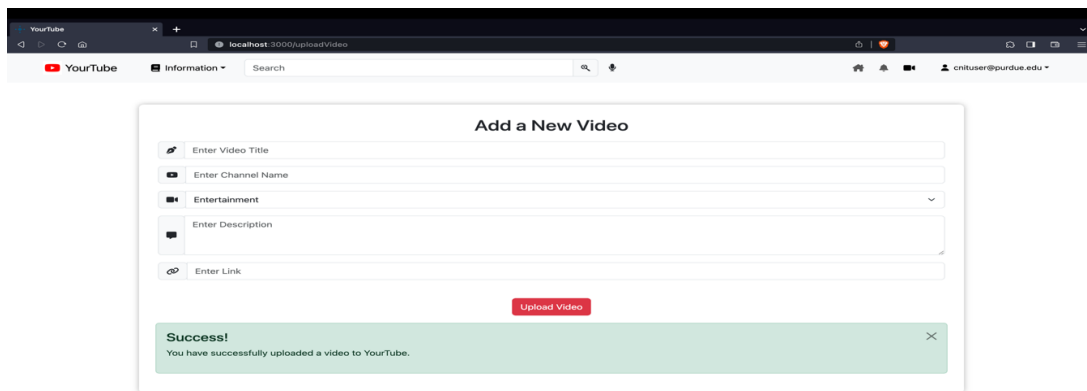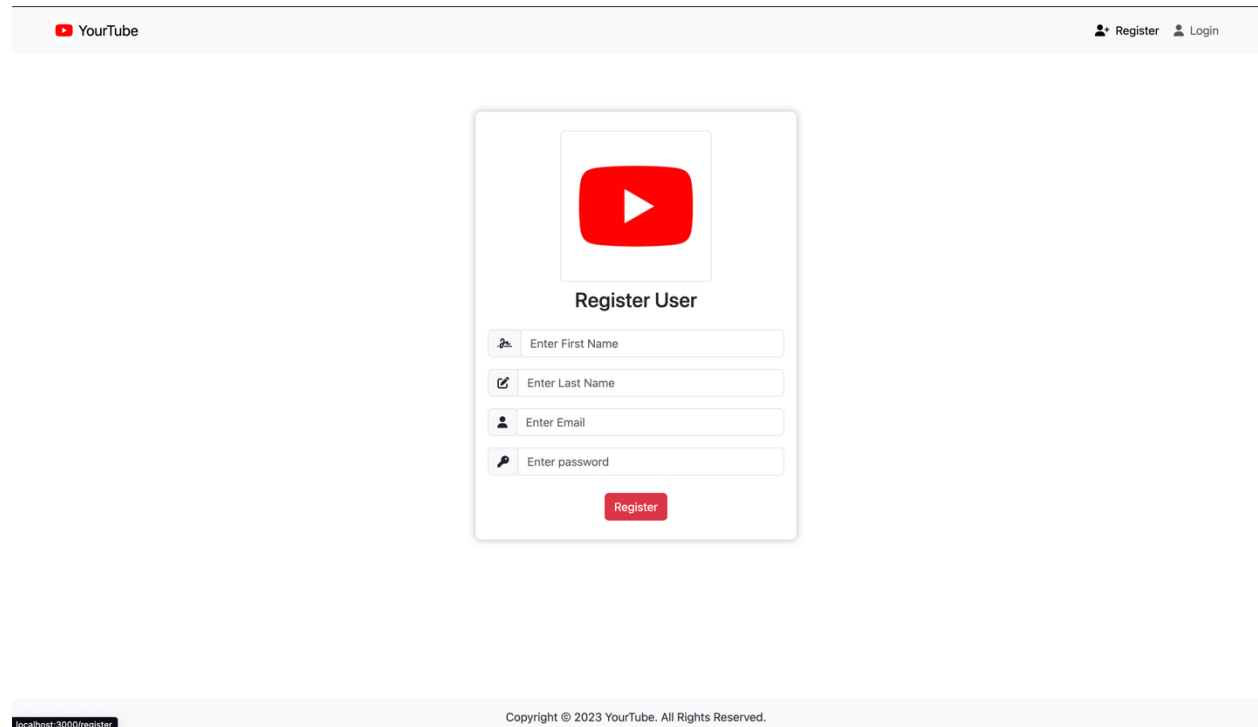
Add View

User can access the video icon on the navbar to add new videos. This operation uses POST method and can be performed only if the user is logged in.

The change can be directly viewed in the home page where a new video will be added based on the addition of new video to the video list from the database.

One other add view is the user registration view (POST method). But the difference is that this add view is an unauthenticated one due to the nature of the action.

The user login is maintained using the Auth Context and local storage in the browser. The backend sends a token (1 hr expiration) when the user successfully logs in. This token can be used for further API calls.

Third-party tools, libraries and frameworks used in the project

1) React.js
2) React Bootstrap
3) React Router
4) Django Rest Framework
5) Django CORS Headers
6) DB - SQLite