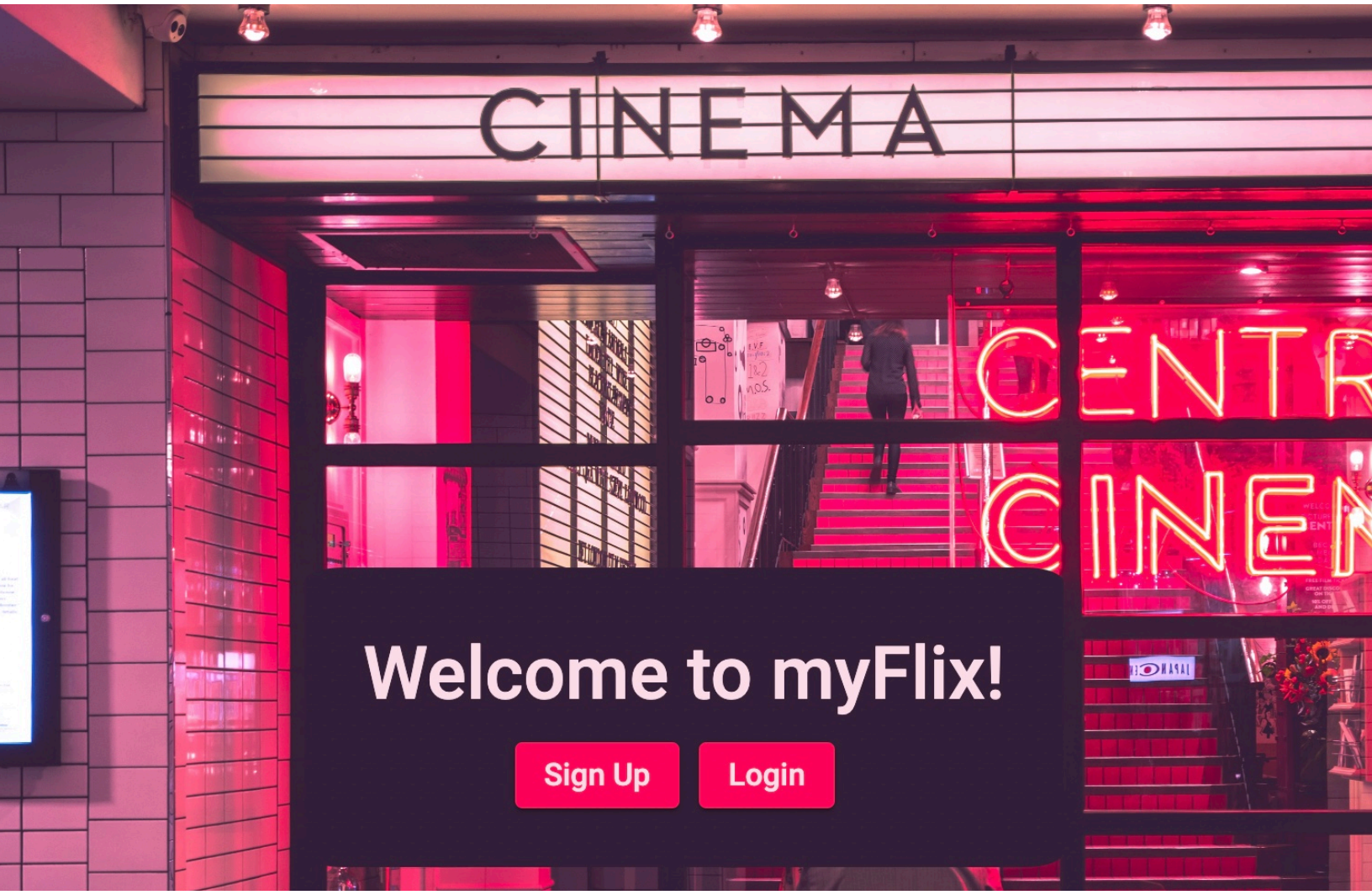
MYFLIX APPLICATION

CASE STUDY



OVERVIEW

MYFLIX IS A WEB BASED APPLICATION BUILT USING THE MEAN STACK (MONGOBD, EXPRESS, ANGULAR, AND NODE.JS). THIS APPLICATION PROVIDES USERS THE ABILITY TO CREATE AN ACCOUNT AND VIEW VARIOUS INFORMATION REGARDING A SELECTION OF MOVIES. THIS INFORMATION INCLUDES MOVIE DESCRIPTIONS, GENRES, AND ITS DIRECTORS. WITHIN THEIR ACCOUNTS THEY CAN SAVE THESE MOVIES TO THEIR FAVORITES, EDIT THEIR ACCOUNT INFORMATION, AND DELETE IT WHEN THEY'RE DONE WITH IT.

GOALS

- BUILD AN APPLICATION FOR USERS THAT ENJOY MOVIES.
- BUILD A RESTFUL API FROM SCRATCH USING NODE.JS AND EXPRESS (THIS WILL HOST THE INFORMATION ABOUT THE INDIVIDUAL MOVIES THAT THE APPLICATION WILL PULL FROM).
- EXPAND, GROW, AND CHALLENGE MYSELF WITH A NEW LANGUAGE I HADN'T USED BEFORE, IN THIS CASE IT'S TYPESCRIPT.

ROLE

- FULL-STACK WEB DEVELOPER

PROJECT SCALE

- 2 MONTH PROJECT

PROJECT SCALE

- CARRERFOUNDRY FULL-STACK IMMERSION COURSE

TOOLS USED

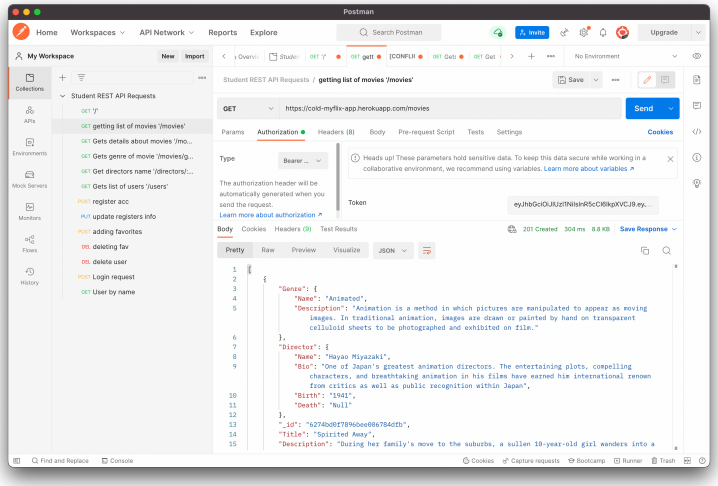
- ANGULAR
- TYPESCRIPT
- MONGODB
- EXPRESS
- NODE.JS
- POSTMAN

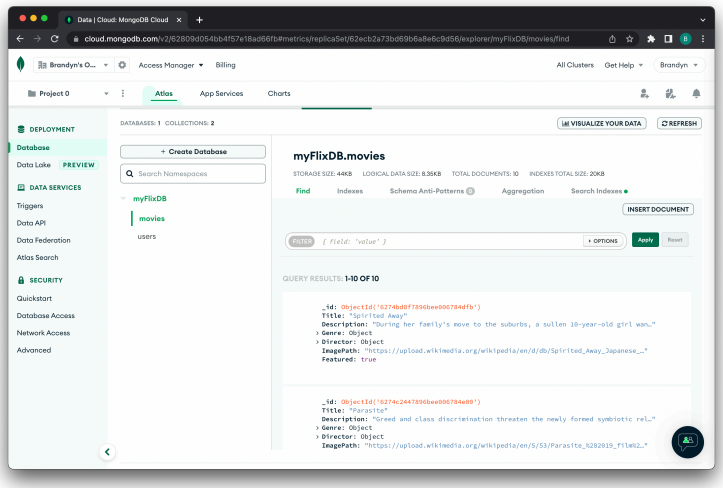



APPROACH

BACK-END

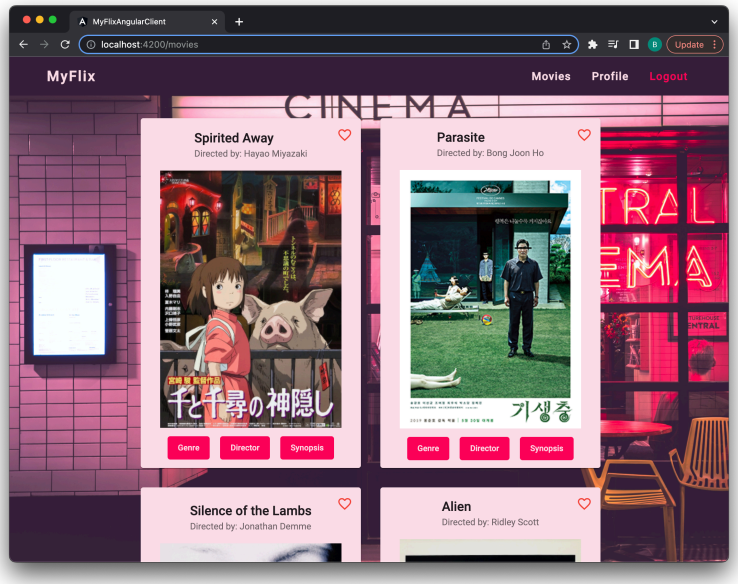
BEFORE I COULD BEGIN THE CLIENT-SIDE OF THE APPLICATION, I NEEDED TO BUILD THE DATABASE THAT CONSISTED OF ALL THE MOVIES WITH ITS INFORMATION. TO DO THIS, I CREATED A RESTFUL API USING NODE.JS AND EXPRESS WHICH I THEN INTEGRATED IT INTO A NON-RELATIONAL DATABASE, MONGODB. TO RETRIEVE THE DATA FOUND INSIDE THE API, HTTP METHODS LIKE GET AND POST, INCLUDING CRUD (CREATE, READ, UPDATE, DELETE) METHODS ARE USED. I MADE SURE THE ALL ENDPOINTS WOULD PULL UP THE CORRECT JSON FORMATTED DATA USING POSTMAN, WHICH IS A VALUABLE TOOL USED TO TEST THESE CRUD METHODS.

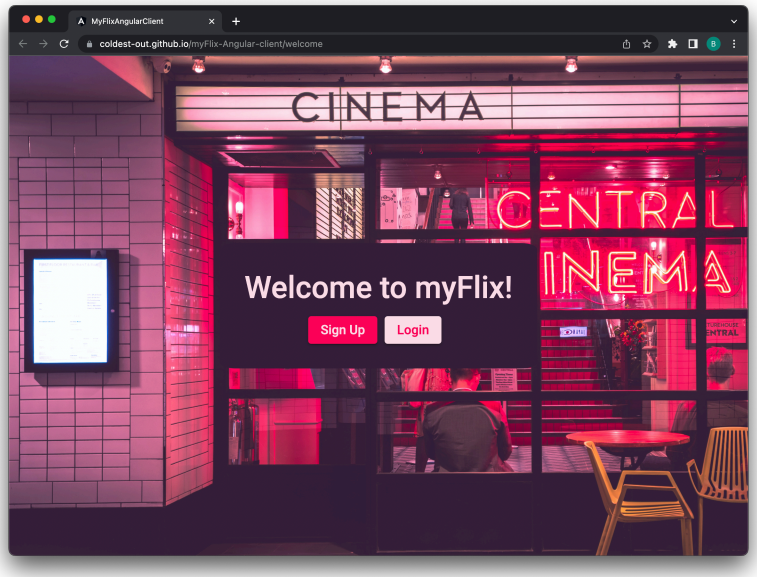




FRONT-END

AFTER THE BACK-END HAD BEEN COMPLETED, IT WAS TIME TO TACKLE THE CLIENT-SIDE UI THAT THE USERS WOULD BE INTERACTING WITH IN ORDER TO RETRIEVE THE INFORMATION. MYFLIX IS A SINGLE-PAGED, ACCESSIBLE, RESPONSIVE APPLICATION THAT WAS CREATED USING ANGULAR & TYPESCRIPT. USING ANGULAR I WAS ABLE TO INTEGRATE A LOGIN & REGISTRATION PAGE WHERE USERS CAN CREATE THEIR ACCOUNTS AND THEN LOG INTO THEM AFTERWARD. I WAS THEN ABLE TO CREATE AN INTERFACE THAT WOULD DISPLAY ALL THE MOVIES POSTER IMAGES ON A MAIN-PAGE, WHERE USERS CAN CLICK ON THEM INDIVIDUALLY FOR FURTHER DETAILS AND ADD THEM TO THEIR FAVORITES. USERS ARE ALSO INTRODUCED TO A PERSONAL PROFILE PAGE THAT WILL DISPLAY THEIR FAVORITE MOVIES THAT THEY CAN GO BACK TOO.





CONCLUSION

DIFFICULTIES & CHALLENGES

THIS WAS AN INCREDIBLY FUN APPLICATION TO BUILD WITH THE TECHNOLOGIES THAT I HAD WORKED WITH. I FOUND MY STRENGTHS WORKING ON THE BACK-END OF THE APPLICATION AND USING NODE.JS, AND I FOUND MY WEAKNESSES WORKING WITH ANGULAR. ALTHOUGH TYPESCRIPT WAS A BRAND NEW LANGUAGE FOR ME TO LEARN, I HAD FOUND SUCCESS THROUGH MY CAREERFOUNDRY TUTOR AND PAIR-PROGRAMMING SESSIONS TO OVERCOME THE CHALLENGES OF LEARNING A NEW LANGUAGE.

FINAL TAKE-AWAYS

AT THE END OF THE PROJECT, I FELT VERY ACCOMPLISHED WITH WHAT I CREATED. I HAD BUILT AN APPLICATION THAT MOVIE LOVERS COULD FIND THEMSELVES ENJOYING. I HAD LEARNED AND OVERCAME A NEW LANGUAGE AND NEW TECHNOLOGIES. ABOVE ALL ELSE, I ADDED MORE SKILLS TO MY ARSENAL AND HELPED FURTHER MY LOVE AND INTEREST IN WEB DEVELOPMENT.