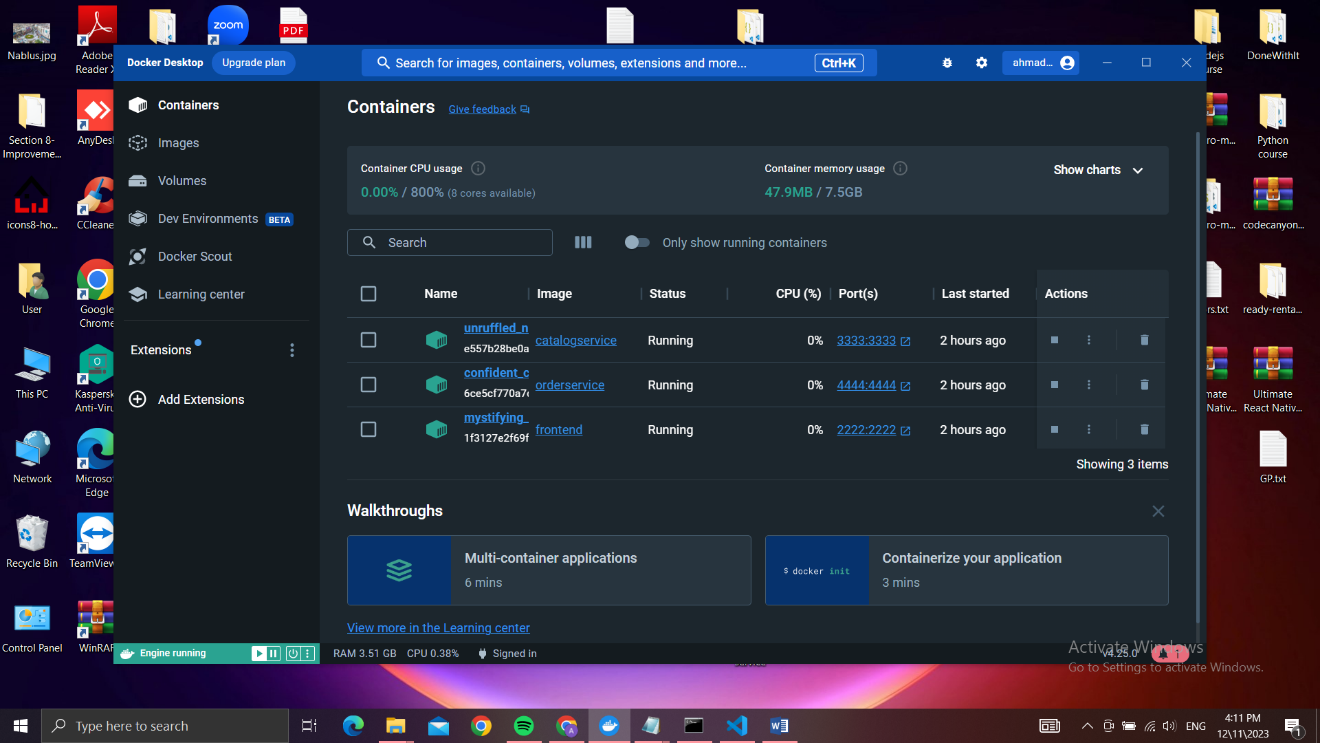
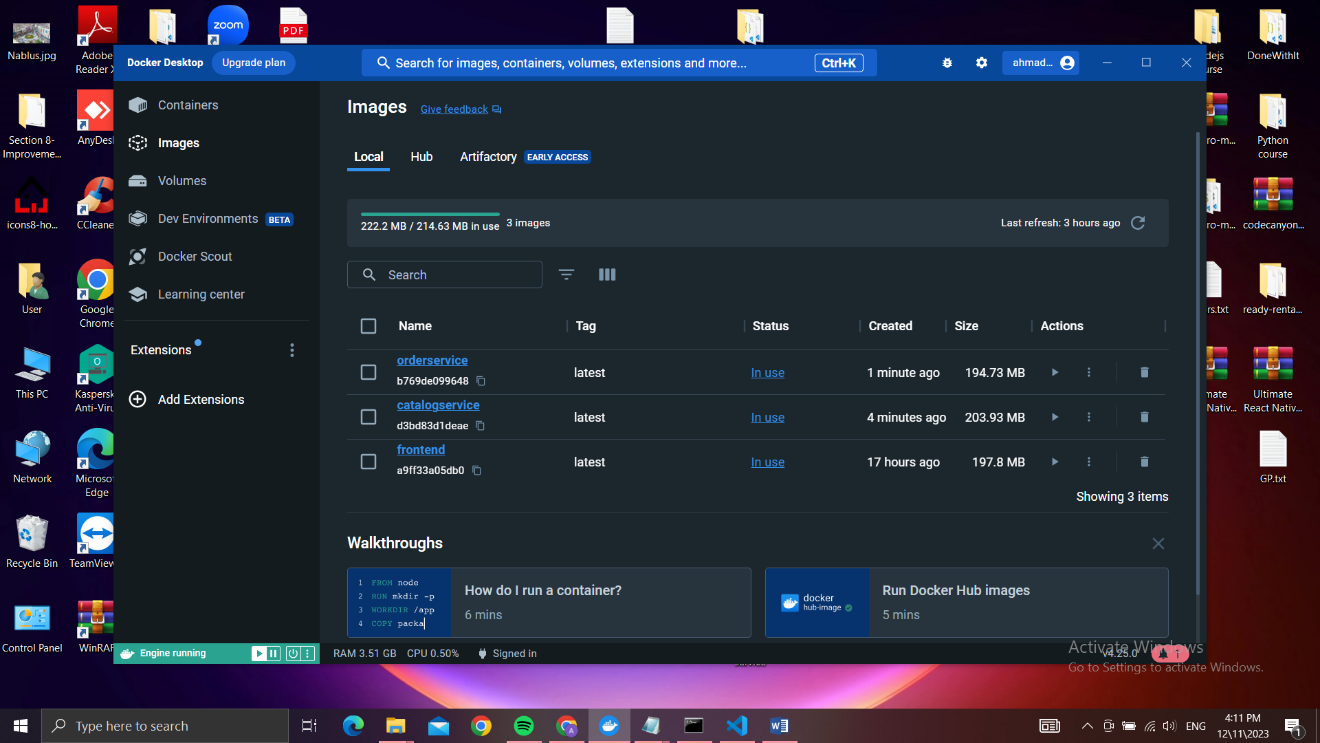
Ahmad Atout

Mohammad Jury

The project was implemented using nodeJS-expressjs as the backend service framework

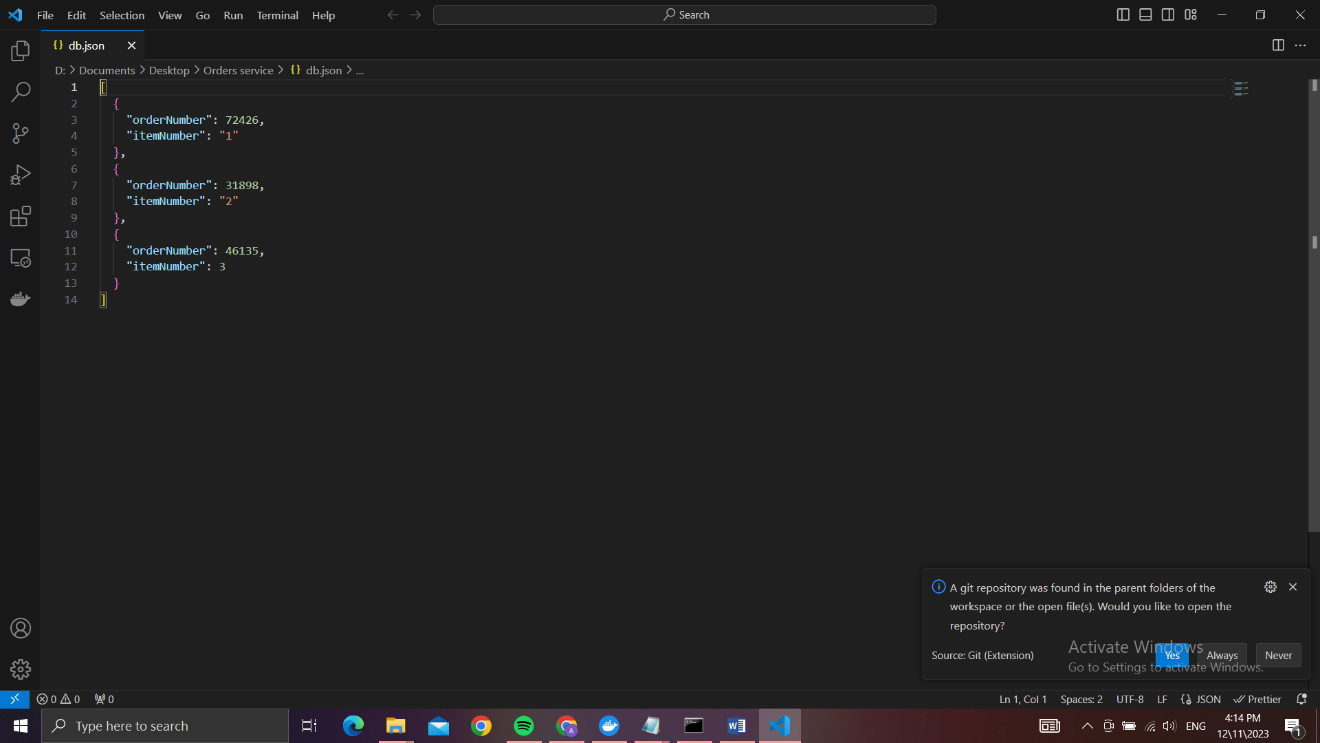
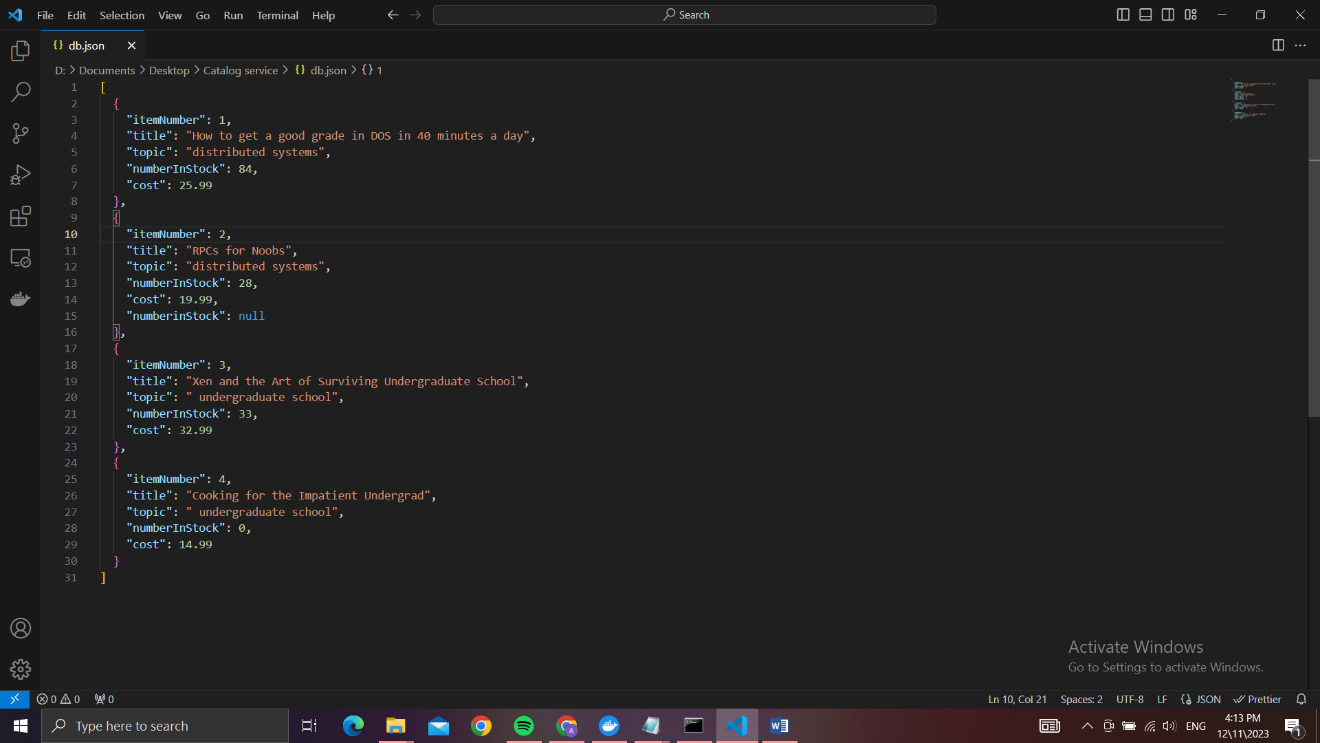
And then every service was deployed on a nodejs docker container



We Used .json files as databases, we had 2 databases each one inside a service

Database for catalog

Database for orders



The catalog service runs on port 3333

The orders service runs on port 4444

The front-end service runs on port 2222

We created Dockerfile in each service to build an image and start a container running that service

FROM node:alpine

WORKDIR /usr/src/app

COPY package\*.json index.js db.json ./

RUN npm install

EXPOSE 4444

CMD ["node", "index.js"]

We used node:alpine because its much lighter version than other node docker images

When an instance is created it runs npm install to install all the required dependencies for the service to run

All the dependencies are mentioned in the package.json file

The requests are directed to the frontend service which then redirects all the requests to the desired service to handle

When the orders service receives a request to purchase it sends a request to the catalog service to query info about the number of the desired book available in stock

If the book is available it sends another request to the catalog server to decrement the number available in stock by 1

How to run our program?

Install each service folder on your computer then run “npm install’ in each folder to install all the required dependencies.

The ip of our local host is hardcoded so change it in each index.js file if needed to your local machine ip address

Run each seavice and test out the project

If you want to run it on different virtual machines

After doing the steps above:-

Bulid an image of each service using the Dockerfile available in each folder of the 3 folders

Then run the images and expose each container the the correct ports

Front-end 2222

Catalog 3333

Orders 4444