READr!

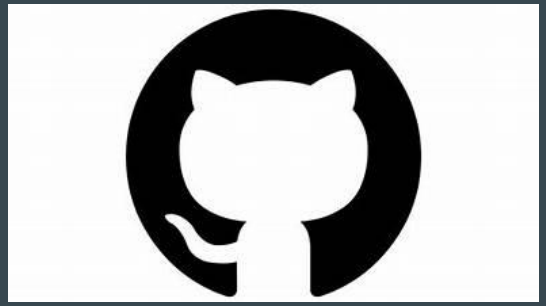
By: Liz Cutting, Rayaan Lodhi, Lucas Fiedler, Bethlehem Belay, Ibrahim Aldulaijan

Description of our project

Our application will be a book tracking interface that allows users to track their own reading progress with insightful statistics, share book reviews with an online community, and explore a collection of books to find their next read.

Tools we used

- Github
- Visual Studios
- Docker
- Azure
- Methodologies
  - Agile methodology
  - Two reviewers for pull request

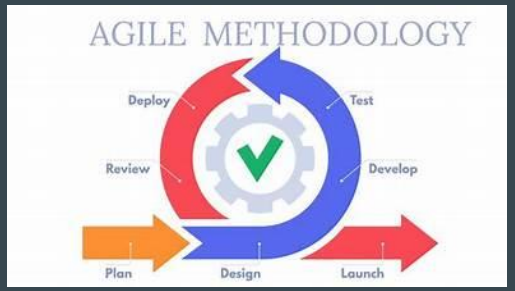


Github was a tool for us to share and update our code with one another
Usefulness: 5/5

VS was a tool for all of us to code in and contribute to the project
Usefulness: 5/5

Used Azure to run application on cloud
Usefulness: 4/5

Docker ran our application using containers
Usefulness 4/5

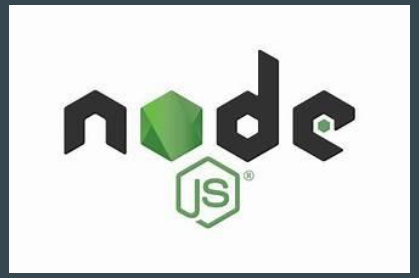


We knew that using Agile will allow us to go back and modify plans
Usefulness 5/5

Expected tools

- HTML/CSS
- NodeJS
- Mocha/chai
- EJS
- PostgreSQL

HTML and CSS was used on the frontend and to design our project
Usefulness 5/5



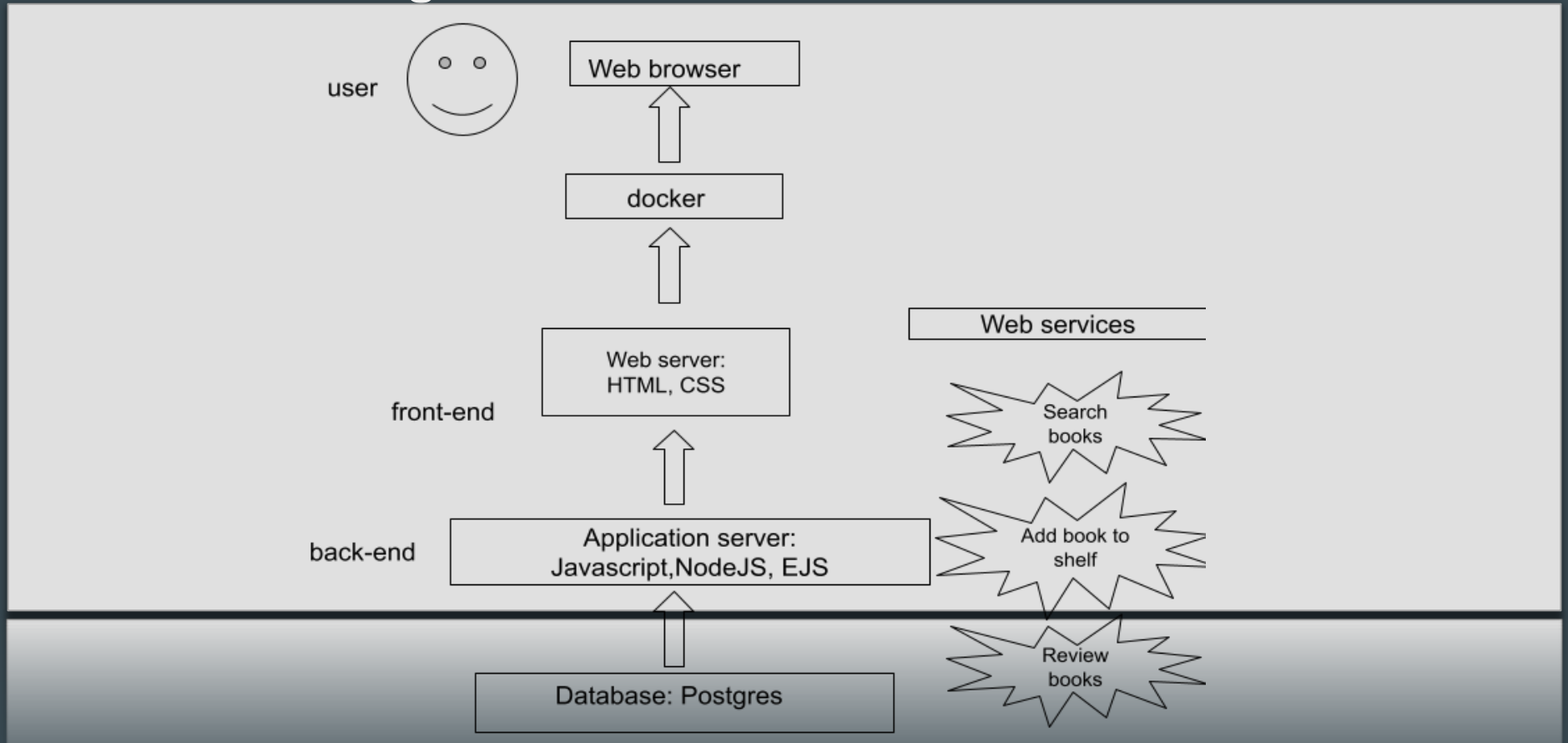
We used NodeJs to run our javascript program on an browser
Usefulness 5/5

Mocha and chai was used to test login page and register
Usefulness: 4/5

EJS was used to dynamically create HTML using user data
Usefulness: 5/5

Postgres was used to store our users datas
Usefulness: 5/5

Architecture Diagram



Challenges

- Making sure users data was saved even after they logged out
  - Solved this issue: by making session variables and solving one small issue we had in the code
- Modal was terrible
  - Trial and error, asking help from peers
- Having to dependent on ones part of the project
  - A lot of communication and making sure we were getting the work done in timely manner to not hold others back
- Getting testing up and running
  - Had to use outside library: sinon