Joshua Hess

Austin, Texas josh@hessjd.com 214-449-8318 https://github.com/DasherDo
https://www.linkedin.com/in/joshua-hess-335526210/

EDUCATION

BS. Biology present
University of Texas, Austin Austin
3.64

SKILLS

Python | **mongoDB/mongoose** | **HTML/CSS** | **JavaScript** (React.js, Node.js, Express,)

Git/Github | **SQL** | **Excel** | **R/R-Studio** | **C++**

PROFESSIONAL EXPERIENCE

Laboratory Supervisor 08/2021 – present
The University of Texas, Austin Austin

- Led and managed a team of 30 researchers over a 2-year period and developed informational materials to support and explain the goals of experiments.
- Trained researchers in the use of R-studio, Excel, and imageJ, enabling them to efficiently and effectively analyze data and complete their research projects.
- Mentored and supported undergraduate students, helping every student achieve success in their research goals and earn a 90% or higher grade.

PROJECTS

Stock Trading Website 2022
Developed a website for buying and selling stock using realtime data

- Developed using MERN stack.
- Created a web-scraping algorithm to obtain current and historical stock prices to provide users with a real-time trading platform.
- Implemented user authentication and integrated a stock data API to transfer data between mongoDB and the frontend.
- Incorporated line charts to show the stock's price over time with interactive features using the Chart.js library.

Realtime Chat App with Socket.io 2022

- Utilized React.js for the front-end interface and a Node.js and Express.js backend to handle server-side processing and routing.
- Incorporated user authentication and utilized mongoDB and mongoose for database management, including storing user information and chat messages.
- Constructed an API to allow the React.js front-end to make requests to the backend and mongoDB, enabling real-time data updates and seamless communication between the front-end and backend.

Sorting Algorithm Visualizer 2021

- Created a web-based React application that demonstrates the steps of various sorting algorithms such as bubble sort, insertion sort and merge sort.
- Utilized interactive visualizations to show the progression of each algorithm and compare them in terms of efficiency and performance.