

MARSHALL CASS

EDUCATION

Full Stack Software Development

devCodeCamp | December 2021

- Utilized JavaScript for algorithm development, coding, debugging and building of functions and graphical user interface applications.
- Successfully incorporated a working introduction of commands, functions and operators used in SQL for extracting data from standard databases.
- Maintained principles and practices of software development, including critical thinking, problem solving skills and essential programming logic.
- Learned fundamentals of internet application design, development and deployment using Django and client-side script languages.
- Built portfolio website using HTML and CSS with own wire framework templates.
- Utilize Django and C# to develop API frameworks with own Entity Relationship Diagrams.
- Utilize Postman to test API endpoints for easy use in front end applications.
- Utilize teamwork, pair coding via zoom, slack, and github.



Marshall.Cass@Live.com



<https://www.linkedin.com/in/marshall-cass-21a1748b/>



(717)489-2871

<https://github.com/MarshallCass>

Technologies

HTML

CSS

JavaScript

React.js

Python

C#

PHP

JSON

MySQL

Django REST Framework

ASP.NET Core Web API Web Services

PROJECTS

Student Management

December 2021

Technologies:

Django REST Framework API, React.js, MySQL,
Postman, HTML/CSS

- Utilized Django REST Framework API to create a database in MySQL that for student and teacher profiles with contact information, grades, assignments, lunch account balances and user authorization
- Applied python in creating methods in Django API, to include get all, filter, post, put and delete methods
- Postman was used to test API endpoints and confirm that methods retrieved appropriate data
- Utilized React.js to create a front-end application to allow users to interact with Django API
- JWT tokens were utilized in async AXIOS functions in React.js to login and retrieve user data linked to the appropriate roles and permissions.
- State components are used in React.js for registration and in the app component, which links the user data to functional components through props
- In upcoming iterations of the project there are plans to expand searching, improve lunch accounting auto recharging account, as well as, the ability to upload completed assignments to provide better communication with parents on how students work is progressing.

https://github.com/MarshallCass/Cap_Backend
https://github.com/MarshallCass/Capstone_Frontend

eCommerce Sports Memorabilia

November 2021

Styling Lead
Technologies:

C# API, React.js, MySQL, Postman, HTML/CSS

- C# is used to create a database and methods for users, products, shopping cart, categories and ratings
- React.js is utilized to allow users to login , log out, add to cart, edit items in cart and check out
- The products in react.js product list component can be filtered in a search bar by category or name
- This project was a group of four all remote, implementing a lot of teamwork, communication, pair programming, and individual tasks with deadlines.

https://github.com/SMay318/Sports_Memorabilia_Back_End

https://github.com/MarshallCass/Sports_Memorabilia_Front_End

Trash Collector

October 2021

Technologies:

Django REST Framework API, Python, MySQL, Postman, PayPal API, HTML/CSS

- Django REST Framework API is used to create methods to create employee and Customer applications
- Created methods in customer application to schedule recurring pickup days, schedule one-time pickups, suspend service, see account details and pay through PayPal. Corresponding HTML files were created in templates to enable use of the methods
- Python was used to create methods in the Employee application to see all weekly pickups by day of the week, as well as today's pickups
- Methods and corresponding html templates enable employees to see Today's pickups, filter out any suspended accounts and add any one-time pickups scheduled for that date. They also allow employees to confirm pickup, which will then add a pickup charge to the clients' balance

https://github.com/MarshallCass/Trash_Collector