MEACHA TAFA

BACKEND DEVELOPER

PROFILE

I'm a Backend Developer with an engineering background, focusing on creating seamless web applications. I bring a fresh perspective to the world of web development with my experience in structural design. My expertise includes Server-Side Development, Database Design, and API Development. I'm passionate about crafting user-friendly web and mobile experiences and committed to making ideas a reality.

With skills in Django, Node.js, and Express.js, I excel in designing efficient APIs and databases. I collaborate closely with frontend teams to enhance user experiences. I'm dedicated to delivering top-notch solutions, combining clean code with creative problem-solving. Let's work together to elevate your projects.

CONTACT

PHONE:
+251-909068750

WEBSITE:
https://meacha1.github.io/My_portfolio_webpage/

EMAIL:
meachattd@gmail.com

HOBBIES

Playing with my kids
Listening to music
Working out in gym
Solving code challenges
Swimming

EDUCATION

Holberton School
Full Stack Software Engineering | Back-end
August, 2022 – October, 2023

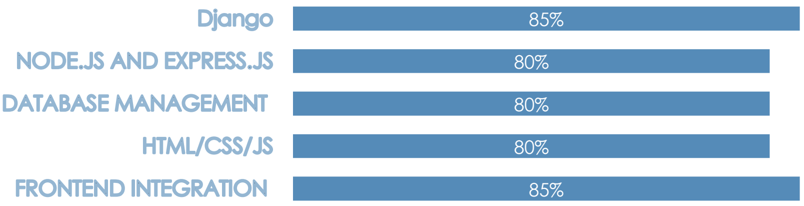
Bahir Dar University, Ethiopia
MSc in construction technology and management
December ,2015- March 01,2018
GPA 3.5

Bahir Dar University, Ethiopia
BSc in civil engineering
September 26,2007- June 22,212
GPA 3.42

WORK EXPERIENCE

Tatkin Consult plc | Structural engineer | Manager
October, 2014 – August, 2022
I have held multiple roles within the company, including manager, structural engineer, project manager, supervisor, and contract administrator. Additionally, I had the privilege of being one of the co-founders of the company.

SKILLS



PORTFOLIO PROJECTS

1. Addis CAD marketplace

DESCRIPTION:

The Addis CAD Marketplace is a CAD-focused e-commerce platform that connects sellers and buyers in the field of Computer-Aided Design. This project provides a wide array of features, catering to user authentication, file management, product listings, payment processing, and more. It aims to offer a seamless experience to users looking to buy or sell CAD files.

KEY FEATURES:

User Authentication and Profiles:

- Registration and login functionality.
- User profiles for buyers and sellers, allowing them to manage their information, purchases, and listings.

File Upload and Management:

- Ability for sellers to upload CAD files in various formats (e.g., DWG, rvt, doc, xls, etc.).
- File management system to organize uploaded files.
- Support for large file uploads.

Search and Filters:

- Search functionality enabling users to find CAD files based on categories, tags, or keywords.
- Filters for refining search results based on file type, price, and popularity.

Product Listings:

- Individual product pages displaying detailed information about CAD files.
- Images, descriptions, file format details, and price information.
- Option for sellers to provide 3D previews or sample images of the CAD files.

Rating and Reviews:

- Ratings and reviews for both buyers and sellers.
- Feedback system to build trust between users.

Subscription Tiers:

- Implemented a subscription-based model with both free and VIP tiers.
- Free tier users have access to basic features, while VIP subscribers enjoy premium services, including advanced market analytics and personalized support.

User Dashboard:

- Dashboard for sellers to track sales, earnings, and customer interactions.
- Dashboard for buyers to manage orders, downloads, and preferences.

Security and Trust:

- Secure payment processing using SSL encryption.
- User data protection and privacy features.
- Verification process for sellers to establish trustworthiness.

Responsive Design:

- Mobile-friendly and responsive design to ensure a seamless experience across devices.

Admin Panel:

- Backend admin panel for site administrators to manage users, listings, and site settings.
- Content management system to update static pages and site content.

TECHNOLOGIES USED:

- **Frontend:** React, Redux for state management.
- **Backend:** Django, MySQL for data storage.
- **Authentication:** Djoser for user authentication.

- **Payment Integration:** Custom mobile money payment integration.

PROJECT CHALLENGES:

- Integrating custom mobile money payment processing.
- Ensuring a responsive design for a consistent user experience across devices.

LEARNING OUTCOMES:

The Addis CAD Marketplace project demonstrates my proficiency in building complex web applications that combine frontend and backend technologies. I have gained valuable experience in e-commerce development, secure payment processing, and user interaction. Additionally, this project showcases my ability to work with various technologies and ensure a smooth user experience.

FUTURE IMPROVEMENTS:

- **Shopping Cart and Checkout:** Enhance the checkout process with streamlined steps and personalized recommendations to increase conversion rates.
- **Messaging System**: Implement real-time notifications for messages and orders, ensuring seamless communication between buyers and sellers.
- **SEO and Marketing**: Optimize SEO strategies, focus on targeted marketing campaigns, and harness social media platforms to drive more organic traffic and boost product visibility.
- Expanding payment options for international users.
- Enhancing search functionality with advanced filters and sorting options.
- Continuously improving user authentication and security features.

CONCLUSION:

The Addis CAD Marketplace project is a testament to my expertise in full-stack web development, offering a complete and user-friendly solution for the CAD community. I'm excited to further develop and expand this project to meet the needs of CAD professionals and enthusiasts worldwide.
https://github.com/Meacha1/Addis_CAD_marketplace

2. Addis Cost Estimation Web Application

OVERVIEW:

In my portfolio project, I developed a dynamic web application called Addis Cost Estimation, showcasing a combination of skills in frontend development, backend programming, and database management. This web application is designed to cater to construction professionals and enthusiasts, providing real-time market data, a seamless user experience, and comprehensive project management tools.

KEY FEATURES:

1. Real-time Market Data:
   - Implemented live data fetching mechanisms to provide real-time price updates for construction materials.
   - Utilized APIs and data scraping techniques to ensure the data is current and accurate.
2. Database Management:
   - Designed and maintained a MySQL database to store user profiles, project details, and market data.
   - Implemented database schema, ensuring efficient storage and retrieval of information.
3. User Registration and Authentication:
   - Created a secure user registration and authentication system.
   - Integrated password hashing and salting techniques for enhanced security.
4. Subscription Tiers:
   - Implemented a subscription-based model with both free and VIP tiers.
   - Free tier users have access to basic features, while VIP subscribers enjoy premium services, including advanced market analytics and personalized support.
5. Bill of Quantities (BOQ) Generator:
   - Developed a BOQ generator tool allowing users to itemize and estimate costs for major construction materials.
   - Integrated smart algorithms for automatic calculations based on user inputs.
6. Customized Project Information Form:
   - Created a user-friendly form for submitting project details.
   - Implemented dynamic form fields, adjusting based on the type of construction project selected by the user.
7. Payment Integration:
   - Integrated a customized mobile money payment gateway for VIP subscriptions.
   - Ensured secure payment processing and seamless user experience during the payment flow.
8. Frontend Development (Pug, CSS):
   - Implemented responsive and intuitive user interfaces using Pug templates and CSS.
   - Focused on user experience and aesthetically pleasing design for all devices.
9. Backend Development (Node.js, Express):
   - Built a robust backend using Node.js and Express.js to handle API requests and database operations.
   - Implemented server-side validation and error handling for user inputs.

TECHNOLOGIES USED:

- Frontend: Pug, HTML, CSS, JavaScript
- Backend: Node.js, Express.js
- Database: MySQL, Sequelize ORM
- APIs: Customized API for real-time market data, Payment API integration
- Version Control: Git, GitHub

OUTCOME:

The Addis Cost Estimation web application showcases my skills in full-stack web development, database management, API integration, and user experience design. It offers an interactive platform for construction professionals to make informed decisions and estimate project costs accurately. The project not only demonstrates technical proficiency but also highlights my ability to create practical solutions for real-world problems.
https://github.com/Meacha1/portfolio-project

3. Basketball Point Counter

PROJECT OVERVIEW:

I created a basketball scoreboard web application using HTML, CSS, and JavaScript. This simple application allows users to keep track of the scores for both the home and guest teams. Users can increment the scores by 1, 2, or 3 points based on the buttons clicked.

SKILLS DEMONSTRATED:

- **Languages:** HTML, CSS, JavaScript
- **Techniques:** DOM manipulation, event handling
- **Concepts:** Web development fundamentals, user interface design

PROJECT DETAILS:

1. Title: Basketball Point Counter
2. Description: The Basketball Point Counter is a web application designed for keeping track of scores during a basketball game. It provides a user-friendly interface where users can easily update the scores for both the home and guest teams. The application features buttons for adding 1, 2, or 3 points to each team's score, and the scores are dynamically displayed on the screen.
3. Features:

- Separate sections for the home and guest teams.
- Buttons for adding 1, 2, or 3 points to the respective team's score.
- Real-time score updates without page refresh.
- Simple and intuitive user interface for easy scorekeeping.

4. Technologies Used:

- **HTML:** Used for structuring the web page and creating the layout.
- **CSS:** Styled the elements, providing a visually appealing and responsive design.
- **JavaScript:** Implemented the functionality for updating scores and handling user interactions.

5. Preview: *[Include an image or GIF showcasing the project]*
6. Project Challenges:

- Ensuring smooth user experience and real-time score updates.
- Implementing a responsive design for various devices and screen sizes.

7. Learning Outcomes: Through this project, I gained hands-on experience in DOM manipulation and event handling using JavaScript. I improved my understanding of creating interactive web interfaces and gained practical knowledge in front-end web development.
8. Future Improvements:

- Implementing a timer or countdown feature for game duration.
- Adding a reset button to reset the scores to zero.
- Enhancing the visual design for a more polished look.

10. **Conclusion:** Creating the Basketball Point Counter was a valuable learning experience that honed my web development skills. It allowed me to apply my knowledge of HTML, CSS, and JavaScript to build a functional and interactive web application. I look forward to implementing more features and continuing my journey in web development.

https://github.com/Meacha1/scoreboard

4. Simple Blackjack Game

PROJECT OVERVIEW:

I developed a simple Blackjack game using HTML, CSS, and JavaScript. This interactive web application allows users to play Blackjack against the computer. The game provides an intuitive user interface for drawing cards, keeping track of the player's score, and determining the winner.

Skills Demonstrated:

- **Languages:** HTML, CSS, JavaScript
- **Techniques:** DOM manipulation, event handling
- **Concepts:** Game logic, user interface design, random number generation

Project Details:

1. Title: Simple Blackjack Game

2. Description: The Simple Blackjack Game is a web-based version of the classic card game, Blackjack. In this game, users play against the computer, aiming to get a hand value as close to 21 as possible without going over. The application provides a realistic card deck simulation, allows players to draw cards, and displays the player's score. The game also determines the winner based on Blackjack rules.

3. Features:

- Start the game by drawing two initial cards.
- Draw additional cards to improve your hand.
- Keep track of the player's score and cards drawn.
- Determine the winner based on Blackjack rules.
- Interactive and user-friendly interface.

4. Technologies Used:

- **HTML:** Structured the web page and created the game layout.
- **CSS:** Styled the elements for an attractive and responsive design.
- **JavaScript:** Implemented game logic, card drawing, and user interaction.

5. Preview: *[Include an image or GIF showcasing the project]*

6. Project Challenges:

- Implementing the game's logic, including card values and determining the winner.
- Ensuring a responsive and visually appealing design for an enjoyable user experience.

7. Learning Outcomes: Through this project, I gained practical experience in JavaScript, particularly in DOM manipulation and event handling. I also honed my problem-solving skills by implementing complex game logic. This project demonstrates my ability to create interactive web applications.

8. Future Improvements:

- Implementing a betting system with virtual chips.
- Adding more features such as splitting and doubling down.
- Enhancing the user interface with animations and graphics.

9. Conclusion: Creating the Simple Blackjack Game was a rewarding experience that allowed me to apply my web development skills to a fun and interactive project. I look forward to expanding the game's features and continually improving my web development abilities.

https://github.com/Meacha1/Blackjack

5. Simple Shell Implementation in C

PROJECT OVERVIEW:

I developed a basic Unix shell in C programming language. This project involved creating a simple command-line interface that allows users to execute commands, manage processes, and interact with the operating system using a custom shell.

Skills Demonstrated:

- **Language:** C Programming
- **Concepts:** Process management, system calls, file I/O, string manipulation
- **Techniques:** Input parsing, command execution, error handling

Project Details:

1. Title: Simple Shell Implementation in C

2. Description: The Simple Shell Project is a command-line interface written in C that mimics the functionality of a Unix shell. Users can enter commands, which the shell processes and executes. The shell supports basic features such as executing external programs, handling input/output redirection, and managing background processes.

3. Features:

- Parse and execute user-entered commands.
- Support for basic shell built-in commands.
- Input and output redirection (e.g., **<**, **>**, **|**).
- Ability to run processes in the background.
- Error handling for invalid commands and system calls.

4. Technologies Used:

- **C Programming Language:** Implemented core functionality and system calls.
- **Unix/Linux Environment:** Developed and tested the shell in a Unix-like operating system.

5. Project Challenges:

- Parsing user input and handling various command formats.
- Managing processes, including background processes and input/output redirection.
- Ensuring the shell's stability and handling edge cases.

6. Learning Outcomes: Through this project, I deepened my understanding of low-level programming concepts in C, including system calls, processes, and file operations. I honed my problem-solving skills by addressing challenges related to command parsing and process management. This project showcases my ability to work with system-level programming and demonstrates my proficiency in the C language.

7. Future Improvements:

- Implementing advanced features such as signal handling and job control.
- Enhancing error messages and providing more informative user feedback.
- Optimizing the code for efficiency and performance.

8. Conclusion: Creating the Simple Shell Project was a valuable experience that allowed me to explore the intricacies of system programming in C. It improved my understanding of operating system concepts and equipped me with practical skills in command-line interface development. I am enthusiastic about further enhancing this project and delving deeper into systems programming.

https://github.com/Meacha1/simple_shell

6. Airbnb Clone

PROJECT OVERVIEW:

I developed a comprehensive Airbnb clone using a stack of technologies including Python, Flask (a micro web framework), HTML, CSS for styling, Jinja for template rendering, MySQL as the database management system, and SQLAlchemy as the Object-Relational Mapping (ORM) tool. This project involved creating a fully functional web application that mimics the core features of Airbnb, including user authentication, property listing, booking management, and more.

Skills Demonstrated:

- **Languages:** Python, HTML, CSS
- **Frameworks:** Flask, Jinja
- **Database:** MySQL
- **Tools:** SQLAlchemy (ORM)
- **Techniques:** Web development, database modeling, user authentication, template rendering

Project Details:

1. Title: Airbnb Clone with Python, Flask, HTML, CSS, Jinja, MySQL, and SQLAlchemy

2. Description: The Airbnb Clone project is a feature-rich web application that replicates the core functionalities of the popular accommodation platform Airbnb. Users can create accounts, list properties for rent, search for properties based on various criteria, view property details, and make bookings. The application provides a seamless user experience with an intuitive interface and real-time interaction with the database.

3. Features:

- User authentication and authorization system.
- Property listing with details such as price, location, amenities, and images.
- Search functionality allowing users to filter properties based on location, price range, and other preferences.
- Booking management for users and property owners.
- Responsive and visually appealing design for various devices.

4. Technologies Used:

- **Python:** Backend logic and server-side scripting.
- **Flask:** Web framework for routing, handling requests, and responses.
- **HTML/CSS/Jinja:** Frontend development and template rendering.
- **MySQL:** Database management system for storing user and property data.
- **SQLAlchemy:** Object-Relational Mapping for seamless database interaction.

5. Project Challenges:

- Designing an efficient database schema to store user profiles, property information, and booking data.
- Implementing user authentication and ensuring secure user sessions.
- Creating dynamic HTML templates using Jinja for rendering data from the backend.
- Optimizing database queries for improved application performance.

6. Learning Outcomes: This project significantly enhanced my proficiency in full-stack web development. I gained expertise in building robust web applications, handling user authentication securely, and interacting with databases using SQL and SQLAlchemy. Additionally, I honed my skills in frontend development, creating visually appealing and responsive user interfaces.

7. Future Improvements:

- Implementing payment gateway integration for secure transactions.
- Adding user reviews and ratings for properties.
- Enhancing the search algorithm for more accurate property matching.
- Implementing real-time notifications for booking updates.

8. Conclusion: Developing the Airbnb Clone was a challenging yet rewarding experience that showcased my ability to work with a variety of technologies in a cohesive manner. It demonstrated my skills in both frontend and backend development, emphasizing my capability to create complex, interactive, and visually appealing web applications. I look forward to incorporating more advanced features and further refining my skills in web development.