

# William Wendt

Long Beach, CA 90712

562-316-9759 | [williamwendt31@outlook.com](mailto:williamwendt31@outlook.com)

Github: [github.com/williamwendt31](https://github.com/williamwendt31)

Portfolio: [william-wendt.tech](http://william-wendt.tech)

LinkedIn: [linkedin.com/in/william-wendt-/](https://linkedin.com/in/william-wendt-/)

## SUMMARY

- Equipped and eager to tackle tough challenges to continue evolving as a developer building innovative web applications with increasing proficiency in *Python, Java, MEAN and MERN* stacks with the ability to develop from scratch.
- Completed 5 projects creating websites in eCommerce, Social Media, Chat room, and Single-Page Web Application Design using *MVC / MVT* architecture and implementation of appropriate technology stacks for responsive design and scalability in an AWS cloud environment.

## EDUCATION

**California State University, East Bay**

**Aug 2018**

*B.S. Computer Science*

Notable Courses:

- Programming Language Concepts | Analysis of Algorithms | Web Site Development
- Database Architecture | Software Engineering | OOP & Design

**Coding Dojo (Certificate)**

**Dec 2018**

- An immersive 14-week program, completing over +1200 hours of hands-on coding in Java, Python and MEAN / MERN stacks. *(Earned perfect scores in all three stacks.)*

## TECHNICAL SKILLS

**Programming Languages:** JavaScript, Python, Java, C++, TypeScript, HTML5, CSS3

**Frameworks:** Express.js, Angular 6+, Django, Flask, Spring Boot, Materialize, Bootstrap

**Servers:** Apache Tomcat, Node.js, NGINX, Unicorn, PM2

**Dev Tools:** VS Code, Spring Tool Suite, NetBeans, Linux Terminal, Postman, MySQL Workbench, Git, AWS

**Methodologies:** OOP, MVC, Responsive Design, WebSockets, AJAX, RESTful API

**Databases:** MySQL, MongoDB, SQLite

**Libraries:** React.js, Socket.io, jQuery, Mongoose

## PROJECT EXPERIENCE

**Individual Consultant - Long Beach, CA**

**Jan 2019 - Present**

*Full Stack Developer*

**AW Pacific Trading Co.**

**Current**

A large-scale e-commerce web app for the consumer and producer. Consumers will have the power to browse, search and purchase items. Implementing a shopping cart in order to keep track of users' selected items. Admin UI allows the producer to have full control to manage products and orders while being able to analyze purchase activity.

- Designing and implementing an open-source, single-page web application following the *MVC* architecture with a responsive UI design while striving for the best UX.

Technologies: *Node.js, Express.js, React.js, Redux, Bootstrap, MySQL, AWS, PM2, NGINX, Git*

**Los Alamitos Lock**

**Current**

An appealing and responsive landing page. The UI is designed to be responsive, easy to navigate and user-friendly. Aiming to offer the best UX possible.

- Constant communication with client to bring their vision to life. Responsive UI implemented using *HTML5, CSS3, jQuery and Materialize*.

**Project: [Chalk It Up](#)****Jan 2019**

A chat app that allows users to communicate with each other through chat rooms. Users can send messages, change their usernames and create/destroy chat rooms.

- Developed an open-source, single-page web application from scratch following the *MVC* pattern with a responsive UI design using *HTML5*, *CSS3*, *Bootstrap*, *jQuery*, and *Angular 7*.
- Server-side engineered with *Express (Node.js framework)* for its robust set of features and flexibility. The server and client communicate and exchange JSON data through *WebSockets (Socket.io)* for instant messaging and real-time updates.
- Stored and manipulated data from chat rooms through *Mongoose / MongoDB* due to its data representation (*JSON*), fast access and scalability. Deployed application with *AWS*, *PM2* and *NGINX*.

*Technologies: Node.js, Express.js, Angular 7, Bootstrap, jQuery, Socket.io, Mongoose, MongoDB, AWS, PM2, NGINX, Git*

**Coding Dojo – Burbank, CA****Sep 2018 - Dec 2018**

*Jr. Full Stack Developer*

**Project: [PetNation](#)****Dec 2018**

A pet adoption site allowing users to find all pets, domestic or exotic, a home. Users have the power to put pets up for adoption, offering all information to potential owners or to adopt any pets they see on the website.

- Built an open-source, single-page web application following the *MVC* pattern to divide the application into three interconnected parts with responsive UI development utilizing *HTML5*, *CSS3*, *Bootstrap* and *React*.
- Server-side engineered with *Express (Node.js framework)* for its robust set of features and flexibility. Connection between the server and client established through *WebSockets (Socket.io)* for full duplex communication and lower latency interaction, which allows for real-time updates.
- Stored and manipulated user data through *Mongoose / MongoDB* due to its high performance and data representation (*JSON*). Deployed the application with *AWS*, *PM2* and *NGINX*. Connected to *AWS S3* to save/load assets.

*Technologies: Node.js, Express.js, React.js, Bootstrap, Socket.IO, Mongoose, MongoDB, AWS, PM2, NGINX, Git*

**Project: [229<sup>th</sup> | 229<sup>th</sup> Admin](#) (email: user | pwd: user)****Nov 2018**

An eCommerce site where consumers can browse the site looking for their favorite products ranging from clothes, to footwear and accessories. Implemented a shopping cart in order to keep track of consumers' desired products integrating with *Stripe API* to complete their transactions. Also, it incorporates an admin UI back-end where authorized users can manage orders and products.

- Designed and implemented an open-source site with *MVC* architecture with responsive front-end development using *HTML5*, *CSS3*, *jQuery* and *Materialize*. Utilized *jQuery* with *Materialize* to handle user interactions and object animation. Used *AJAX* to send requests to the server to query the *SQLite* database for searching and sorting.
- Used *JSON* data to make requests to the *Stripe API*. Utilized *Django-ORM* to create, read, update and delete user-data in *SQLite*. Configured and set up a virtualized environment with *AWS*, *Gunicorn* and *NGINX*. Connected to *AWS S3* to save/load assets.

*Technologies: Python, Django, SQLite, Stripe API, jQuery, AJAX, Materialize, AWS, Gunicorn, NGINX, Git*

**Project: [Roots](#)****Oct 2018**

A social-media website allowing users to upload and share user-generated content. Users can interact with their uploads by either commenting, liking, or disliking while allowing interaction with other users by following them or sending them direct messages. *Roots* allows for creative posts and creative profiles.

- Engineered an open-source website with *MVT* architecture using responsive UI development with *JSTL*, *HTML*, *CSS3*, *jQuery* and *Materialize*. Implemented a sorting feature for user generated content and real-time data loading with *AJAX*. Used *jQuery with Materialize* to handle user interactions and object animation.
- Programmed the logic and data layers using *Java*, *Spring framework*, *JPA*, and *MySQL*. Configured *Spring Boot Security* for authentication and authorization. Deployed the application with *AWS* and *Apache*. Connected to *AWS S3* to save/load assets.

*Technologies: Java, Spring Boot, Spring Boot Security, MySQL, jQuery, AJAX, Materialize, AWS, Apache, Git*