Tanner Velten

Los Angeles, CA
tannervelten@gmail.com
(805) 710-6704

linkedin.com/in/tannervelten
github.com/tannervelten
tannervelten.com

TECHNICAL SKILLS

Strong: React, JavaScript, HTML, CSS, Node, Meteor, Heroku, Git, Mongo, JSON, NPM, Promises
Experienced: Bootstrap, Express, SQL, C++, SCSS, PostgreSQL, React Router, Express Router

PROFESSIONAL EXPERIENCE

HateCrimeMap **UCLA AISC/IDRE** - GIS Web Developer — Feb 2017 - Present

- Designing and developing web application using GIS software to visualize harassment claim data.
- Integrated React front end UI with Express back end and deployed to Heroku.
- Built Express endpoints and data models to route data from a PostgreSQL database.
- Utilized React to create custom, responsive web components and Jest + Enzyme for unit testing.
- Set up cron job script to automate URL validity checks and flag invalid URLs in the database.

Tutorfly - Interim CTO, Full Stack Software Engineer — May 2017 - Dec 2017

- Implemented custom React components designed with modularity and reusability in mind.
- Reduced file sizes in UI codebase by 50% using functional components and React best practices.
- Restructured front end file structure to support feature-first organization and clean up code.
- Managed team of 6 software engineers and assigned weekly tasks using agile methods.

Nopical - Software Engineering Intern — Jan 2017 - May 2017

- Constructed back end methods using Meteor to securely expose MongoDB data to the client.
- Increased site security by adding user confirmation checks to back end Meteor methods.
- Built server-side filtering methods to prevent sensitive data from being published to the client.
- Migrated codebase to be compatible with updated versions of Meteor for improved performance.

SOFTWARE PROJECTS

Bird Scooter Petition - React, Meteor — Bird Scooter Petition

- Built full stack application with React + Meteor, hosted by Heroku with mLab for cloud storage.
- Implemented presentational React component to reactively update when new data is inserted.

Battleship - C++ — Source Code

- Built classes using OOP, inheritance, and polymorphism to keep program modular and extensible.
- Implemented recursive ship placement algorithm to keep code clean and enemy AI fresh.

Treasure Hunter - JavaScript — Treasure Hunter

- Puzzle requiring the troll in a dark cave to build a staircase up to the gold at the top of the tower.
- Implemented iterative DFS path-finding AI for the troll to quickly solve the puzzle.

Tanodoro Clock - JavaScript — Tanodoro Clock

- Implemented timing algorithm using JavaScript timing events to execute JS at specific intervals.
- Created intuitive UI to handle timer functions for an object oriented data model.

EDUCATION

University of California, Los Angeles - Los Angeles, CA — Expected Spring 2018

- B.S. Cognitive Science
- Coursework in Computer Science and Program in Computing