STEPHEN PARK

BSA COMPUTER SCIENCE | 2017

Phone *832.370.5690*
Web *stephensp.com*
Email *stephenspark@utexas.edu*
Github *github.com/ssp0929*
Codepen *codepen.io/ssp0929*

EDUCATION

The University of Texas at Austin
B.S.A. Computer Science
Business Foundations Program (BFP) Certificate
2013 - 2017 | 3.10 GPA

Relevant Coursework
Data Structures, Algorithms & Complexity
Discrete Mathematics, Linear Algebra
Probability & Statistics
Computer Architecture, Operating Systems
PHP & SQL, Compilers
Computer Networks, Wireless Networks
Computational Brain

SKILLS

Languages
JavaScript (ES6), Python, C++, HTML5, CSS3, SQL, JQuery, PHP, Bash

Frameworks / Libraries
React, Node.js, Express, Vue, Angular (2+), Axios, D3.js

Technologies
Git, Linux, Windows, Heroku, AWS, PostgresQL, MySQL

PROJECTS

Auction Website (HTML, CSS, PHP, MySQL)
Class project aimed to simulate a real client website design workflow, with simulated client interaction and feedback in between sprints. Front end was built using HTML and CSS. Back end was built using MySQL and PHP. Website was hosted on a university Apache server. Users were able to create accounts, bid on auctions, create auctions, and a payment model was mocked up in place. Inputs were santized, HTTPS was enforced, and security measures were implemented against XSS vulnerabilities, SQL injections, and malicious binary image data execution. Website is currently deployed on Heroku.

React Auction Web Application (HTML, CSS, JavaScript, React, Node.js, Axios, Express, PostgresQL)
Refactored my LAMP stack auction website using a more modern tech stack and designed to operate as a single page application. React was the framework used for the front end, Axios used in middleware to make HTTP requests from the front-end client to the front-facing API created using Postgrest to allow user interaction with the back-end database. Node/Express/PostgresQL were used to build the back end. Website is currently deployed on Heroku.

Incremental Game (HTML, CSS, JavaScript, Angular(2+), Node.js, Express)
JavaScript-based game created to be played in browser based on the same mechanics as the currently trending “idle” games found in web apps and mobile apps. Instead of opting for traditional back-end storage, I have decided to use localStorage to allow for persistent data storage without significant overhead.

Pascal Compiler (C)
Project for Compilers class. Compiles a subset of the Pascal codebase. Implemented part of a lexical analyzer, parser, and code generator. Several tools were utilized: YACC/Bison and Lex.

Discord Bot (Python)
Created a python-based bot that interacts with users in a discord group. The bot periodically makes an API call to a REST API, parses the JSON data, and allows users to make queries on that data. Additionally, the bot also has other commands that users can call. The bot is hosted on my RaspberryPi.

Please check my github and codepen for other projects.