Andy Phan

□ aphan1@cs.washington.edu | 1 (360) 489-4941
 □ aphan1.pages.cs.washington.edu/personal-website/
 in https://www.linkedin.com/in/andy-phan-cs/

EDUCATION

University of Washington

Seattle, WA

Bachelor's degree in Computer Science

Expected March 2022

GPA: 3.72

South Puget Sound Community College

Olympia, WA

Associates of Arts degree

Graduated June 2019

GPA: 3.95

EXPERIENCE

Freelance Web Developer

Summer

 Designed and constructed mobile-friendly websites for clients using HTML, CSS, JavaScript, and Node.js 2020

Prepared multiple designs and templates for client approval before development

SKILLS

Programming Languages: Java, Python, HTML/CSS, JavaScript, SQL, Lua

Frameworks/Tools: Node.js, React.js, Bootstrap, Git, MySQL, SQLite, Firebase, Heroku

PROJECTS _____

 Participated in Hack'20 online hackathon as a team of four Co-built React web texting app that pairs clients to others based on a struggle they may have so they can vent anonymously Implemented user authentication and a real-time database using Google Firebase Python, Pygame Built a Python app that displays animations for various maze generation and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation SpotifySort Personal Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 	React.js,	OpenBar Hackathon	Summer
Summer Pygame Python, Pygame Built a Python app that displays animations for various maze generation and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation React.js, Express.js, Bootstrap, Heroku struggle they may have so they can vent anonymously lmplemented user authentication and a real-time database using Google Firebase Summer 2020 Summer Summer 2020 Summer 2020 Summer 2020 Summer 2020 Summer 2020 Summer Express.js, Bootstrap, Heroku Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features	Bootstrap,	 Participated in Hack'20 online hackathon as a team of four 	2020
 Implemented user authentication and a real-time database using Google Firebase Python, Pygame Built a Python app that displays animations for various maze generation and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation React.js, Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 	Firebase,	 Co-built React web texting app that pairs clients to others based on a 	
Firebase Python, Pygame Built a Python app that displays animations for various maze generation and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation SpotifySort Personal Express.js, Bootstrap, Heroku Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features Summer 2020 Summer	Git	struggle they may have so they can vent anonymously	
 Python, Pygame Built a Python app that displays animations for various maze generation and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation React.js, Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 		 Implemented user authentication and a real-time database using Google 	
 Built a Python app that displays animations for various maze generation and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation SpotifySort Personal Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 		Firebase	
 and maze solving algorithms such as Kruskal's algorithm and A* Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation React.js, Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Heroku Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 	Python,	MazeAlgorithms Personal	Summer
 Used Pygame to implement a GUI with customizable maze sizes and step-by-step animation React.js, Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Heroku Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 	Pygame	 Built a Python app that displays animations for various maze generation 	2020
by-step animation React.js, Express.js, Bootstrap, Heroku SpotifySort Personal Output Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features		and maze solving algorithms such as Kruskal's algorithm and A*	
React.js, Express.js, Bootstrap, Heroku SpotifySort Personal Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features		 Used Pygame to implement a GUI with customizable maze sizes and step- 	
 Express.js, Developed a React app using the Spotify Web API that retrieves the client's playlists and sorts the songs based on the client's selected option Heroku Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 		by-step animation	
 Bootstrap, Heroku Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 	React.js,	SpotifySort Personal	Summer
 Rendered data from AJAX calls to an Express.js backend to display playlists, songs, and their audio features 	Express.js,	 Developed a React app using the Spotify Web API that retrieves the 	2020
playlists, songs, and their audio features	Bootstrap,	client's playlists and sorts the songs based on the client's selected option	
	Heroku	 Rendered data from AJAX calls to an Express.js backend to display 	
Lug CDC Standard Totric Dorsonal Winter		playlists, songs, and their audio features	
Luu, Sh3-Stailualu lettis Feisoliai Willer	Lua,	SRS-Standard Tetris Personal	Winter
Love2D • Built video game clone of Tetris implementing many modern-day Tetris 2019	Love2D	 Built video game clone of Tetris implementing many modern-day Tetris 	2019
features including animations, SRS-standard rotation system, custom		features including animations, SRS-standard rotation system, custom	
controls, and random sequence generator.		controls, and random sequence generator.	