

Nathan Garrett

nate.garrett1992@gmail.com
www.nathanpaulgarrett.com

214-809-9502 | Allen, TX
www.linkedin.com/in/nathan-garrett
github.com/nateusmc

Expert RPA and Automations Developer and Full Stack Web/Mobile Developer experienced with React, Node.js, mobile-first responsive development, and business process efficiency and automation. Skilled with modern tooling and eager to learn more.

KEY SKILLS

Certifications: AWS Certified Cloud Practitioner, Microsoft's ProcessRobot (Now Microsoft's Power Automate) Process Development Diploma as well as their Operations and Administration Certification

Languages/Frameworks: RPA Development, React, Redux, JavaScript, Node.js, SQL, Python, HTML5, CSS3, MongoDB, Express.js, Ajax, as well as multiple unit test and assertion libraries (Mocha, Chai, Jest, etc)

Dev Tools & Methods: Agile, Multiple AWS Services, Git/GitHub, Mobile-first, Responsive, Test Driven Development, Big O Notation, Data Structures and Algorithms, CI/CD Pipelines

Online Marketing: SEO/SEM Optimization, Google Analytics, Campaign Targeting/Management, Web Design

Other: ProcessRobot - Process Development Advanced Course Certified, Process Robot Operations and Administration Advanced Course Certified

EXPERIENCE

RPA Developer - Contingent at John Deere Financial, 3/2021 – Current

- Increased overall company productivity and efficiency by developing and maintaining 11 attended automations currently being used in production with extremely minimal maintenance and over 99% up-time. These automations are implemented across multiple departments within John Deere Financial and collectively save close to 500 hours of otherwise manual work per month as well as increasing job satisfaction for the employees that used to do these processes manually and eliminating potential errors when previously performed by employees.
- Assisted in the training and continued skill development of a group of citizen developers internal to John Deere Financial that has resulted in more automations being used in production saving time and minimizing human error.
- Currently working with higher-ups to design and implement an Automation Catalyst Curriculum/Training to further empower our current (and future) citizen developers to be successful at scale.
- Designed and implemented a scalable infrastructure for RPA development to scale over time by designing a database for Dev, Cert, and Prod that gives RPA flows increased intelligence as well as the ability to self-heal from previous failed runs without human intervention and an error logging system for better runtime monitoring. This will drastically reduce on-going development hours and maintenance on all RPA processes across the board.
- Built an end-to-end database-driven framework/template, complete with error handling, to further strengthen the scalable infrastructure described above. This will ensure quality, consistent processes are being built by developers and citizen developers alike as well as speed up time spent on future development and debugging substantially.
- Was able to cut company costs by eliminating the need for third-party contractors in some cases by developing an RPA flow that replaced their existing workloads.
- Built a large RPA solution which worked with the SAP desktop application that previously had an automation built using VBScript. I was not only able to make this automation easier to maintain, but also increased consistency and efficiency and solved some edge cases that the previous solution was not able to handle, further improving the overall process.
- Improved business process quality (both with and without the need for RPA flow(s) to be built) when I discovered a better way to fix/modernize the existing business processes themselves. On a few occasions this included finding bugs in code within other department's tools/websites that they were previously unaware of.
- Worked with Microsoft's Power Automate team on numerous occasions to both enhance the tool we use to build RPA flows as well as work out bugs that I've discovered while working with it.
- Was a presenter in large scale meetings (1,000+ attendees) to raise awareness of RPA automation and the benefits we've seen since starting to utilize RPA across John Deere Financial.

▪ **RPA Developer - Big Data Energy Services, 10/2019 – 2/2021**

- Increased overall company productivity and efficiency by developing and maintaining an in-house ETL Framework, RPA libraries, and SQL table-driven architecture for use with RPA processes.
- Enabled the company to quickly grow to meet new client demands by training 3rd-party consultants to build some RPA processes for us as well as training some in-house devs from other departments within the company
- Single handedly built 20+ RPA processes end-to-end for Shell Natural Gas along with many more processes for other big name clients (100+ RPA processes in total)
- Increased Operations department productivity many times over by architecting a queue and re-scheduling system and designing systems for our RPA processes to self-heal without manual intervention (often being able to recover even after being up to several months behind). This drastically reduced man hours and maintenance on all RPA processes across the board (over 1,400 RPA processes running daily)
- Assumed the role/job duties of the Vice President of Development after he stepped down back in 12/2019

Full Stack Web and Mobile Developer - Tracker Technologies, 4/2018 – 9/2019

- Mobile App Showcase: <https://www.rev1energy.com/the-rev1-app>
- Built several components using React to use in conjunction with existing SaaS web applications.
- Built some Node.js services that drastically increased process efficiency and cut process runtimes from multiple hours down to under 18 seconds per run. These services directly impacted our end-users in a positive way.
- Increased overall company efficiency as well as cut down huge monthly costs not having to pay for 3rd-Party ATS systems anymore by building an in-house ATS system from the ground-up. I was the sole developer on this project and was responsible for it end-to-end. It was based on BullHorn ATS design, but with better UI/UX and cut out some features that were just clutter for our company when using bullhorn. This included a desktop app for our HR department to use as well as complimentary iOS and Android apps where candidates and clients could submit their interest in positions to the HR department.
- Created services for task automation using Node.js and xBasic (a proprietary backend language used by Alpha Anywhere Software)
- Redesigned entire UI/UX for our main product, TrackerDB, that oil and power service contractors used out in the field to document when tests were performed on machinery, keep track of maintenance schedules, and any other commissioning needs. These UI's came with multiple differing versions that could be swapped out dynamically based on which client was logged into the system. This assisted in unique designs per client and essentially white-labeled our software.

Full Stack Developer, 4/2017 – 3/2018

- During this time I had the opportunity to work with such languages and technologies as: JavaScript, Python, React, Redux, MongoDB, PostgreSQL, multiple test and assertion libraries, and even socket.io to implement real-time chat.
- Collaborate several hours every week with a senior web developer to studied new languages and frameworks.
- Create fully functional applications while working on a team in an Agile environment.
- Worked with multiple groups of developers (usually groups of 5) to build many different types of web applications. Some groups we developed in an agile manner, others we took a Test Driven Development approach and everyone on the team got great experience on both the back-end and front-end of these projects. We utilized Travis CI for our CI/CD pipeline.

Mood Cloud

- Technologies used: React, Redux, MongoDB, Node.js, Express.js, Jest, Mocha, Chai, Flexbox, Socket.io
- Server and hosting Technologies: Node server as well as deployment via Heroku, Netlify and continuous integration with Travis CI
- This app was designed to reduce the amount of bullying in today's school system and give the student a more comfortable place to communicate the issues that are occurring and help teachers recognize these issues and resolve them in a timely manner.
- Users can login as either a student or a teacher. As a student they are able to login and submit their daily emotions from a list of 100 options to choose from. These emotions are then documented inside a word cloud showing at a glance the emotions that most frequently. They also have the ability to ping their teacher whose class they've enrolled in and have a live 1 on 1 real-time chat with them. Teachers also have their own dashboard allowing them to see submitted emotions of all students enrolled in their class as well as participate in a real-time 1 on 1 chat with a student in crisis.

Kurikaeshi (Japanese Learning)

- Technologies used: React, Redux, MongoDB, Node.js, Express.js, Flexbox
- Server and hosting Technologies: Node server as well as deployment via Heroku, Netlify and continuous integration with Travis CI
- This app was created to use a spaced repetition algorithm and singly linked list to create a web based Japanese language learning application. As the user answers questions throughout the app they will be assigned strength values according to how many times and how frequently they got the question wrong causing it to appear more often and help them learn it.

Additional Work History

Salem Carriers | Delivery Driver, 10/2016 – 8/2017

GATX Consulting (internet marketing and consulting agency) | Founder, 4/2015 – 1/2017

XXL HOT SHOT TRANSPORTING LLC | Owner/Operator, 7/2015 – 11/2015

Acme Truck Line, Inc. | Owner/Operator, 8/2014 – 7/2015

Performance Food Group | Delivery Driver, 2/2014 – 8/2014

United States Marine Corps | Active Duty (Honorably Discharged), 8/2012 – 9/2013

EDUCATION

THINKFUL | Web Development Intensive, 2/2018

- Intensive, five-month full time course on modern full stack JavaScript development.
- Learn industry best practices and practical software development standards.
- Focus: HTML5, CSS3, JavaScript, jQuery, Node.js, Express, Ajax, React, Redux, and algorithms & data structures.