**Will Midgette**

**Full Stack Developer**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Toronto, ON** | **(647) 523 8234** | [**wamidgette@gmail.com**](mailto:wamidgette@gmail.com) | [**GitHub**](https://github.com/wamidgette) | [**LinkedIn**](https://www.linkedin.com/in/will-midgette-5399a3189/) | [**Portfolio**](https://willmidgette.com/) |

**Professional Summary**

Self-motivated full stack JavaScript developer with a background in engineering and project management. Experience developing scalable, data-driven web applications with Node.js, React and other languages and frameworks. Passion for providing aesthetic and functional software solutions to business problems.

**Main Technical Skills**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Front End | | | Back End/Other | | |
|  |  |  |  |  |  |
| * CSS | * SCSS | * HTML | * Node.js | * SQL | * PHP |
| * JavaScript | * Responsive Design | * React | * Git | * REST APIs | * AWS EC2 |

**Experience**

|  |  |  |
| --- | --- | --- |
| **Full-Stack Developer**, **Pliteq Inc.** |  | February 2022 to Present |

Hired as the sole web developer to maintain and develop custom web applications and improve the efficiency of general business and manufacturing processes. Communicate effectively with multiple teams across company with different levels of technical backgrounds to develop software solutions to unique business problems.

* Design and develop internal full-stack Node.js and React.js applications integrated with NetSuite ERP using Oauth2 protocol to streamline business processes.
* Deployment of all company web applications using Git and AWS EC2 Linux servers.
* Project managed the migration of the company’s maintenance management system (CMMS) to a new platform. Generated requirements documents and guided new platform’s developers through the integration process with NetSuite. Cut yearly CMMS costs by nearly 50% through successful migration.
* Reduced production reporting errors by over 80% by improving error tracking on web application used by shift supervisors. Production errors that do occur are now more likely to indicate actual problems in product BOMs, which would otherwise go unnoticed.
* Implement new features in Laravel application requiring full stack code changes and database migrations. Security features have led to improved capability of admins to track activity of users and protect company-critical data.
* Develop custom NetSuite ERP scripts to automate company processes and eliminate pain points for everyday system users.
* Developed two 3PL EDI integrations with trading partners using Boomi middleware. Greatly reduced the need for email communication between logistics teams and our 3rd party warehouses.

|  |  |  |
| --- | --- | --- |
| **Full-Stack JavaScript Developer Intern, Bird&Be** |  | October 2021 to January 2022 |

* Developed product recommendations webpage from start to finish using React.js functional components, and API methods within Next.js framework.
* Implemented Swiper and PhotoSwipe APIs to create carousels and modal image galleries for products.
* Developed efficient API methods for fetching data from BigCommerce. Executed requests in parallel wherever possible to reduce data retrieval and page load times. Used dev tools to measure efficiency improvements.
* Wrote concise CSS code to reproduce mockups with grid, flexbox, and animations. Factored repeated styling into re-usable classes and reduced number of media queries for responsive design.
* Collaborated with design team, advising on development restrictions while proposing alternative solutions to design problems.

|  |  |  |
| --- | --- | --- |
| **Freelance API Developer, JBox Creative** |  | August 2021 to October 2021 |

* Created custom WordPress API endpoint for hotel rental company allowing them to update all hotels on their website daily through a single POST request.
* Developed function to convert request body data to JSON and archive data at dynamically routed locations for future reference.
* Developed function to update custom fields and taxonomies of WordPress posts from JSON data – when a post does not exist in the database, a new one is automatically created and populated.

**Education**

|  |  |  |
| --- | --- | --- |
| **Cert., Web Development, Humber College** |  | September 2020 to September 2021 |

* One-year web development graduate certificate. Focus on building fundamental skills in PHP, JavaScript, MySQL, HTML, and CSS.

|  |  |  |
| --- | --- | --- |
| **B.Eng, Civil Engineering & Co-op, McMaster University** |  | September 2013 to April 2019 |

* Five-year engineering and co-op degree program focused on structural and geotechnical engineering.

**Projects**

**NetSuite Photo Upload Application**

* Problem: Dozens of large bags of raw material brought into factory daily, all with different spec sheets. Specification documents were being physically adhered to products as they progressed through manufacturing process. Risk of losing entire history of the manufacturing life cycle of finished goods if documents were lost.
* Solution:
  + - * Part 1: React.js application hosted on NetSuite as a page available to production workers. User enters a work order number, which automatically starts a media stream through the USB camera on the device. User lays out their documents in the camera frame, then clicks “Capture Image.” This pauses the video stream and takes a snapshot of the frozen image using the html canvas element. Image sent as Base64 encoded URL string to custom NetSuite RESTlet endpoint. String converted to .jpg file and attached to NetSuite work order for future reference.
      * Part 2: A second React.js static build application which allows users to search by products by unique code, pulling data from the NetSuite ERP and displaying the user a full manufacturing routing history, otherwise unavailable through the ERP’s UI.

[**Explore Cities Project**](https://explore-cities.vercel.app/)– **Helps users evaluate cities based on criteria of their choosing**

* React.js application utilizing Google Maps API, Teleport API, and Chart.js.
* User selects topics of interest for which they would like to see data.
* Google Map object loads, and GET request is sent to Teleport API for all cities in their database.
* For each city returned, Google Map marker is created with Onclick trigger.
* When a city marker is clicked by the user, another GET request is sent to Teleport API for city data based on user’s selected preferences.
* City data is graphed for user in three categories: “Salaries”, “Statistics”, and a general city “Report”.

**To see more of my projects, please visit:** [**willmidgette.com/projects**](https://willmidgette.com/projects)