

Rich Martinez

Contact

richard.i.martinez.jr@gmail.com | [GitHub](#)

Experience

Senior Software Engineer, Quicken Loans

November 2019 - June 2022

- System Design
 - For prioritizing service virtualization: Created a roadmap, and architecture design diagram, worked with stakeholders to create a feature estimation with the Weighted Shortest Job First scale and created the features and stories with well-defined acceptance criteria.
 - Created a TypeScript & Mountebank application of virtualized services, using ECR, ECS, and EKS to represent service dependencies of the Rocket Mortgage Origination frontend.
 - For request logging with Mountebank, the virtualized services integrated with a TypeScript Lambda, behind APIGateway with a defined OpenAPI contract, that accepts Mountebank requests and logs them to an S3 bucket.
 - Created a TypeScript CLI to autogenerate the boilerplate infrastructure, boilerplate TypeScript application for each virtualized service, and CircleCI config updates.
 - Thoroughly documented with many examples covering the why of the application, how to set up new virtualized services, updating existing virtualized services, and infrastructure overview.
- CI/CD Improvements
 - Created a Composer package called code-quality-tools that was a collection of the Rocket Mortgage Origination CI/CD scripts, to be reusable among all Rocket Mortgage Origination CI/CD pipelines. Thoroughly documented readme that provided examples of how to use each script and provided a commonly used template for a collection of composer.json scripts that mapped to the CI/CD script and made it easy to run things locally as well as in the pipelines.
 - Created a merge freeze continuous integration script written in TypeScript that automated blocking pull requests from being merged during a merge freeze.
 - Created a detailed plan to remove the merge freeze requirement and never block merges but instead block automatic deploys during the deployment window.
 - Created thorough documentation for use of the CI/CD script including multiple logic flow diagrams to describe exactly what was happening on run.
 - Created a Node CLI to autogenerate a Postman collection, including pre-request OAuth script, using an OpenAPI contract. To be run as a CircleCI job.
 - Created CircleCI merge workflow jobs to automatically build docker images and push them to internal Artifactory to speed up local development with prebuild images.
 - Reduced the time it took to run CI unit test and generate code coverage by 60%.

- Development LifeCycle Improvements
 - Reduced the number of manual steps needed and reduced OS compatibility issues during setup and build for multiple repositories by managing everything through containers.
 - Setup remote debugging for various repository containers using Xdebug, special docker DNS, and Remote Explorer for Visual Studio Code, and added thorough step-by-step documentation on debugging setup.
- Technology Standards
 - Created a CI/CD standard for Rocket Mortgage Origination frontend repositories including
 - pipeline requirements for unit tests, acceptance tests, e2e tests, mutation tests, performance tests, load tests, SCSS linters, Typescript linters, static analysis, unit test coverage, deploy ability checks, and health checks
 - pipeline optimizations for automatically generating a release at scheduled times and linting commits
 - diagrammed continuous integration and continuous delivery ideal workflows for pull requests, merged code, release tag, and scheduled quality checks
 - Created a security risk mitigation standard for Rocket Mortgage Origination frontend, which included
 - existing Rocket Mortgage InfoSec standards that must be followed
 - tools that must be used to mitigate risk including SonarQube, Snyk, CodeQL, docker-file-image-update, npm-check-update, Helmet, and csrf
 - tools that should be used to mitigate risk including cors and a tool for dynamic application security testing
- Mentoring
 - Mentored junior software engineers on using the development environment of the Rocket Mortgage Origination codebases, object-oriented programming concepts, and design patterns used by the codebases.
 - Answered technical questions, offered to pair program with software engineers to share knowledge, and helped prepare junior software engineers for promotional interviews.

Software Engineer, Quicken Loans

June 2017 - November 2019

- Development LifeCycle Improvements
 - Helped develop, document, and diagram a new standard for the development process, including branching strategy, automated deployment to lower environments, and pull request code quality requirements.
 - Created a TypeScript CLI that prompts the user to associate the quality gate scripts with a user-selected Git Hook (e.g. pre-push).
 - Updated the frontend build step to run tasks in parallel and make all the functions asynchronous.
 - Implemented code splitting to lazy load modules with dynamic imports via Webpack.
 - Updated docker images to LTS versions.
 - Added a cache-buster for the assets to benefit the clients.
- Application Design

- Created a multistep online schedule closing experience BFF API and view modules for Rocket Mortgage Origination clients to close loans faster.
- Built out a co-branded origination experience BFF API and view modules for the clients of Rocket Mortgage Origination partners.
- Created a containerized JavaScript application of virtualized services representing all the dependencies of the Rocket Mortgage Origination frontend application, to speed up development and make the acceptance test more reliable.
- Mentoring
 - Mentored junior software engineers on using the development environment of the Rocket Mortgage Origination codebases, object-oriented programming concepts, and design patterns used by the codebases.
 - Advised the Rocket Mortgage Origination backend team on a plan to transition TFS version control to Git version control, which included transferring commits, branching strategy, Git commands, and merging strategy & code quality checks.

UI Developer, Quicken Loans

August 2014 - June 2017

- Fully Redesign the One Reverse Mortgage frontend
 - Collaborated with the design team to create a One Reverse Mortgage style guide for all the one reverse mortgage products both internal and external.
 - Worked with the design team to fix designs that I considered overly complex UI for the web. Thoroughly encouraged a mobile-first design flow. Pushed back on designs that were bad UX with accessibility in mind.
 - Replaced all image graphics with SVG for scaling consistency. Implement SVG animations with SnapSVG for things like taking a base SVG of the United States map and providing tooltip functionality with information about lending for each state.
 - Implemented BEM ideology for all SCSS.
 - Added better client-side validation and maintained regular expression simplicity through concatenating separate reusable regex variables, for things like year, month, date, phone number, etc.
 - Created a basic auth app for MyORM using express and a JWT package.
 - Fully documented application redesign for things like, installation, running the application, debugging, building, unit testing, writing client-side validation, etc.
- Optimized frontend build process
 - Replaced Browserify with Webpack, using a default Webpack file and multiple environment files that import the default file.
 - Added postcss step for Autoprefixer for browser prefixes and purifyCSS to decrease CSS file size by a few percent, by removing unused classes.
 - Added shrinkwrap to the project to ensure package version consistency across installs, before package-lock was available.
 - Created separate modules for each gulp task and updated all the tasks to use ECMAScript 2015.

- Optimized continuous integration quality by setting a standard to require JavaScript unit tests using Mocha.
- Mentored UI developer interns and junior software engineers
 - Taught them CSS/SCSS best practices and methodologies (e.g. BEM, ITCSS, Atomic CSS), ECMAScript 2015 features, semantic HTML, and some standards from WCAG 2.0.
 - Provided thorough explanations for any suggestions I might have had in reviewing their code.
 - Provided opportunities as needed to pair program.
- Contributed to JavaScript Standards Group
 - Helped create standards for things like client-side frameworks and associated technologies, linting, and BFF layer technologies.

Web Developer Internship, Quicken Loans

May 2014 - August 2014

- Created and managed internal WordPress websites
 - Created custom themes with LESS
 - Added new content areas as needed
 - Fixed invalid HTML for content editors
- Set WordPress site standards for the content management system to avoid unwanted content behavior
 - Disallowed HTML in most content inputs
 - Limited editing capability of the What You See Is What You Get tools to provide only things needed for the content editors

Open Source Projects

Remove Stale Branches CLI - July 2018

- Created a Node.js CLI for automatically removing any number of local and/or remote branches that have been merged upstream.

Skills

Language Experience

- HTML/Twig
- CSS/SASS/LESS
- JavaScript/TypeScript
- PHP
- .NET/C#
- Python
- sh/bash/zsh

Framework/Library Experience

- Angular
- React
- Express
- Next.js
- Symfony
- Slim

Tools

- Git
- CircleCI
- Docker
- Kubernetes/Helm
- Terraform
- Postman
- Splunk/Grafana

Education

Baker College of Clinton Township, MI

- Bachelor of Digital Media Technology - September 2011
 - Major - Digital Media Design

Baker College of Flint, MI

- Associate of Business - August 2009
 - Major - Graphic Communications