Matt McCaffrey

Montclair, NJ | hello@mattmccaffrey.com | https://www.mattmccaffrey.com/

Software industry veteran. Thought leader. Always learning, always on the cutting edge.

Seeking a position in the greater New York City area that would leverage my expert technical and leadership skills to grow with an exciting web-based product or service.

Employment

Full-Time:

Some Spider Studios - Principal Engineer and Architect (2018-Present)

Building software to grow the brands of Some Spider Studios: Scary Mommy, The Dad, Fatherly and Famfare.

Responsible for estimating new projects and turning requirements into technical plans and then code across the full stack spanning multiple web applications, microservices, and APIs. Also responsible for any required DevOps and system administration, including CI/CD pipelines and setting up and maintaining infrastructure on AWS.

Mentored a small team, interviewed and hired an additional developer.

Technologies used include PHP, Laravel, Node.js, PostgreSQL, Redis, Vue, Bootstrap, Ad Management APIs and multiple other REST APIs, AWS (EC2, Lambda, RDS, ElastiCache, S3, IAM, Route53), Firebase and React Native.

Development:

- Built a tool that increased the click-through rate of articles posted on Facebook by allowing editors to generate multiple "bundles" of headline, status copy and photo variations. The system moved a test through a state flow of unlocked/locked/boosting/boost completed/scheduled, where on approval, an ad for each bundle would be automatically generated and then the version with the best CTR would be automatically scheduled to our Facebook page.
- Saved SEO Team ~$30k/mo by replacing a "Link Equity Tool" service that we paid for with our own tool. It was composed of a Laravel administrative backend which allowed rules to be defined to generate recommended articles based on the one being viewed, PostgreSQL and ElastiCache (Redis) as a data store, a client JavaScript library which was included and used on our WordPress sites, and a Node.js Lambda function which made the recommendations when consumed by the client library across our high-traffic sites.

- Built multiple reporting dashboards to pull, gain insights from and visually report on data from Facebook, Instagram, YouTube, Snapchat, Twitter, Pinterest and Google Analytics. These reports were heavily relied on and saved many hours of time for users across departments.
- Built The Dad mobile application which allowed us to schedule push notifications to the 15,000 users who installed the application. The application allowed users to view "Dadisms" and meme images, which were added daily. It was built in React Native, used Laravel as an administrative backend and for the API, and Firebase for the cross-platform notifications and event analytics.
- Built Famfare as an auto-scaling, serverless site and administrative backend using Laravel and Laravel Vapor. In addition to authenticated admin users from our company who could manage Partners, Promotions and Codes, this allowed users from the internet to create a passwordless account, where they could claim limited-release coupon codes for small DTC brands that we partnered with.
- Wrote hundreds of PHPUnit tests.

Stealth Startup - CTO (2018)

Built MVP. Details available on request. Technologies used include AWS, Python, and Django.

LittleThings - Senior PHP Engineer (2016-2018)

LittleThings engaged over 50 million visitors each month and had over 3 billion video views in 2017. On February 28, 2018, LittleThings closed its doors to all 100+ employees.

Responsible for full stack development across five applications. Technologies used were PHP, Laravel, Node.js, Express, PostgreSQL, Elasticsearch, Redis, AWS, Vue, Twitter Bootstrap, Semantic UI, gulp, webpack, Facebook's Marketing API, Facebook's Graph API, and many other REST APIs.

Development:

- Built Odin, a programmatic ads management system which allowed for the generation and tracking of unique UTM sources tied to ad campaigns for URLs on the LittleThings web site. It successfully arbitraged traffic bought from Facebook, Taboola, Revcontent, Yahoo, Outbrain, and Content.ad by using their APIs to monitor costs in relation to revenue generated from the traffic sent. It also automated the creation of Campaigns, Adsets, Ads and Creatives in Facebook, allowed for budget changes to be made to Facebook Ads in bulk, supported a set of configurable event triggers, and made intelligent recommendations for new Ads to be created. It included many hourly and daily reports, graphs and alerts which drove decision-making for the media buying team, allowing them to consistently make profits up until LittleThings closed. It managed over 20,000 UTMs. This was made in Laravel.

- Built Apollo, a video metadata management system which allowed bulk uploading of videos, ingestion of user-approved videos from configurable feeds, and publishing of configurable feeds for the consumption of multiple OTT platforms. This allowed videos from LittleThings to be available on major distribution platforms like Roku, Apple TV, Amazon Fire TV, XUMO, Pluto TV and others. This was made in Laravel.
- Built GeoIP, a scalable, high performance IP to Geolocation API in Node.js using Express. Due to high traffic, LittleThings was paying over $2000/mo to use a 3rd party API for this purpose. This replaced it at a cost of under $100/mo in AWS fees, and it was built as a side-project. It was used to determine where viewers were located, and restrict access to content, especially licensed videos, based on that data.
- Launched many features in ReportingThings, an analytics application that was used by our external Partners and internal Partnership Team to track and report traffic and revenue generation. It managed over 1,500 partners simultaneously. This was made in Laravel.
- Built a trending content discovery engine which aggregated data from multiple sources and calculated a weighted engagement velocity based on the engagement since time published relative to the influence of the publisher. Also calculated the sentiment score as well as trending bigrams, trigrams and word stems.
- Contributed multiple accepted pull requests to Laravel framework, and attended Laracon 2016, Laracon 2017, and 2ndQuadrant PostgreSQL conference 2017.

FortressITX & Solar VPS - Development Lead (2012-2016)

Responsible for all front end and back end development of web-based eCommerce store and custom control panel used by Solar VPS clients to launch and manage their Cloud VPSs, in addition to the internal tools used by Fortress ITX and its brands. This included the full product lifecycle from requirements gathering, requirements writing, coding, testing and launching. Hands-on development and management of a remote team.

Primary technologies used were PHP, a custom-built MVC framework, CodeIgniter, Laravel, MySQL & MariaDB, Twitter Bootstrap, JavaScript, jQuery and other open source JavaScript libraries, HTML, CSS, and JSON/JSONP.

Development:

- Launched and continued to iterate on a custom-built eCommerce store and control panel that allowed purchasing and management of custom configurations of both Linux and Windows VPSs. Integrated WHMCS billing, SolusVM provisioning (Xen PV & Xen HVM), and Kayako support systems.

- Built a "One Click Application Install" platform that allowed clients to preview, install, uninstall and upgrade 260+ different applications on their VPS through an easy to use web interface that leveraged Twitter Bootstrap, jQuery, and Handlebars.js.
- Built an automated test suite in CasperJS to install and display screenshots of all of the installed applications.
- Built "SickBay" feature that allowed clients to purchase virtual credits that could be used in exchange for technical support.
- Built a Facebook-like notification system to alert clients of new events (eg. a ticket was responded to, a new invoice was created) while logged into the control panel, and email them a weekly compendium of any unread notifications.
- Built and open sourced a functional prototype of a mobile application that allowed clients to manage their VPSs. I oversaw the development of a client API and then wrote this as a proof of concept to show what the API could be used for. Used JSONP, jQuery Mobile, and PhoneGap.
- Built a Solar VPS game and leaderboard that was used to attract people to our booth at a tradeshow, with prizes given to winners.
- Built an abuse tracking system that improved and automated the process used by our abuse department to handle complaints about ip addresses that we owned. Included a dynamic UI that resembles Apple's Mail.app, processing of incoming email, auto-categorization of complaint type by context, status workflow with views of different queues, a dashboard with charts and graphs, screenshots of offending urls, audit trails of staff activity, and lots of other neat features.
- Built and open sourced an "Ask the Company" application that allowed staff to anonymously ask questions to management. Included email alerts, an admin backend for publishing answers, and an interface for staff to search the published answers to questions.
- Built support for a custom, rule-based payout structure with the Commision Junction affiliate network. This included integration with our eCommerce store in addition to creating internal reports and tracking that allowed us to monitor referrals. Weekly "batch files" of conclusion and correction data based on order status was automatically generated and sent to Commision Junction each week.
- Integrated all order details with eCommerce Tracking in Google Analytics and Piwik.
- Built a migration tracking system to invite old clients from six different legacy brands in different billing systems to the current Solar System, emailing each a unique coupon code and tracking the conversions.
- Pitched and developed a system for awarding cryptocurrency to clients with purchase.
- Introduced a web-based VNC for Windows VPSs, and web-based SSH for Linux VPSs in our control panel by using open source technologies.
- Exported all SalesForce objects and loaded them into MongoDB as separate collections. Built a front-end for searching and browsing the data, which accessed MongoDB through Ajax calls to a simple PHP script. The front-end was built with AngularJS, Angular UI Grid, and Angular directives for Bootstrap. This

"FortressFORCE" tool allowed us to cancel our SalesForce subscription and switch to a different, empty CRM.

- Wrote a tool that used the Bing API to find web sites hosted by ip addresses that we owned, get their Alexa rankings, and submit their content to Open Calais which used NLP and machine learning to extract structured information from the content. Stored this in a local MySQL database and built a UI for exploring that information so that we could understand more about what our clients were using our services for. As we sometimes got support calls from clients of hosted clients, this tool also helped identify who the client was when receiving calls like this.
- Built a front-end replacement for accessing tickets, contacts and companies in ConnectWise. This improved workflow for our own team, and we also sold it as a service.
- Built a tool for storing ConnectWise client's credit card data in Authorize.net's CIM, browsing invoices from ConnectWise, and capturing payment.
- Oversaw development of and sometimes contributed code to an internal inventory tracking system, a website monitoring system, a client API, a disaster recovery addon that we sold, a system that did daily syncs of clients and services from multiple billing systems to SalesForce, and a shipment tracker that leveraged the Aftership API.
- Wrote a "provisioner platform" and matching storefront that allowed us to pre-rack servers that when ordered, were automatically turned on with IPMI, bootstrapped with an operating system using Kickstart and a Handlebars template, and had networking enabled using our custom networking api. The goal of this project was to allow us to pre-rack multiple servers to sell instead of building and racking them when an order came in. The project was completed but the needs of the business changed (we stopped selling dedicated servers) and it was not launched.
- Assisted in the physical move of 1000+ servers from one data center to another. Shutdown, unracked, moved, rewired and reracked servers in new location. Wrote tools to verify that servers in specified racks were responsive after move.
- Responsible for delivering numerous financial reports and statistics from our different billing systems during Chapter 11 filings.

Management & Process Refinement:

- Introduced version control. Initially installed inhouse Subversion server, then later began using Mercurial Hg hosted on Bitbucket for other projects.
- Introduced code reviews using ReviewBoard.
- Introduced project management, feature planning and bug tracking. Used Redmine.
- Introduced a simple requirements and feature breakdown process using a custom template that I made on Google Docs.
- Introduced new technology and design patterns to development team members.
- Contributor to company blog and knowledge base.

Ecko|Code - Senior Web Developer (2011-2012)

Responsible for PHP and JavaScript development of Dexter: Slice of Life, a Zynga-style Facebook game that coordinated weekly missions in the game based on the latest Showtime episode.

- Improved code maintainability by rewriting backend to use a custom MVC framework and be object oriented.
- Integrated key events in game with Kontagent analytics platform.
- Responsible for all Facebook API integrations such as game installation, credits payments, and posting to walls.
- Built dynamic html email templates to allow for delivery of unique content based on user statistics. Verified compatibility across all email clients with Litmus. Used Amazon SES to deliver emails to over 500,000 addresses, with ability to opt-out of future emails.
- Built system that allowed "codes" to be shared and redeemed for various in-game rewards.
- Built debugging system to snitch any client-side Javascript errors to our server.
- Built internal dashboard that was used by Business Intelligence to push changes in game economy from development to live servers.

Numara Software - Development Manager (2006-2011)

Hands on development and management of developers, features, and releases of FootPrints, a fully web-based, ITIL-compliant service desk, configuration management and change management solution that is delivered and sold B2B as a hosted service, or as a download that customers can deploy on their own.

I originally developed FootPrints as part of UniPress Software, and grew it from a small, two person project to a multi-million dollar suite. Numara Software acquired UniPress Software, where I had worked since 1997, and was then acquired by BMC Software in 2012.

Feature Manager Role:

- Supported developers in implementing new features, including hands-on development
- Technology selection and architecture
- Wrote requirements documents for new features including user interface specification, system requirements specification, and system design specification

- Worked with PM to agree on requirements, socialize feature across departments, guide user research sessions and analyze output
- Assisted QA in developing test plans
- Work breakdown and assignment
- Code reviews using Review Board
- Most recently completed managing Form Designer feature, with hands on development. Form Designer is a drag and drop WYSIWYG form builder that uses YUI3

Release Manager Role:

- Prioritized defects and change requests
- Coordinated defect management with outsourced developers and inhouse team
- Assessed risks and planned mitigations for successfully meeting release dates
- Provided weekly update on release status with stakeholders

Professional Services Point of Contact:

- Gathered requirements from customers for customization requests to FootPrints
- Estimated cost, scope, and timeline for customizations
- Wrote words for statement of work that is delivered to customers

Other Responsibilities & Accomplishments:

- Administrative management of direct reports
- Conducted quarterly and yearly reviews
- Planned yearly and quarterly goals with direct reports
- Received Customer Advocate company award in 2011
- Received Big Kahuna department award in 2008 for being instrumental in FootPrints 8 release
- Blogger on company blog
- Created DevTalk internal blog for freeform discussion and brainstorming among developers

UniPress Software - Development Lead (1997-2006)

One of two original developers that built FootPrints, a fully web-based service desk and issue tracking system that has been featured favorably in the Gartner Magic Quadrant and supports high-volume, enterprise clients. Personally responsible for the design, architecture, and development of much of the product, and was a hiring manager and mentor for most of the team.

FootPrints is written in Perl with a SQL back-end, and utilizes html, css and javascript on the front-end. Versions are available for Unix, Linux, and Windows operating systems and database support is available for MySQL, Microsoft SQL Server, Oracle, and PostgreSQL.

Database Design & Implementation

- Converted FootPrints from using a proprietary GDBM database backend to a relational database and sql
- Tested against and made changes to support popular database platforms
- Wrote installation scripts to properly create user, database, permissions and tables for new installations of FootPrints on each of the supported database platforms
- Documented data model and entity relationships
- Wrote mrXMLDump/mrXMLLoad utility to enable cross-DBMS migration of application data
- FootPrints allows administrators to define custom, dynamic field configurations to define issues and can perform custom, complex searches on issues

Escalation Design & Implementation

- Escalations allow workflow, automation, and business intelligence rules to be created in a simple web based administration screen
- Criteria ("if this") define triggers
- Actions ("then that") define what happens when criteria is met
- Macro language allows FootPrints issue data to be passed to external applications

CMDB Design & Implementation

- CI Types can be configured with different custom attributes per type
- Relationship Types can be defined with constraints on what CI Types can have that relationship
- Service Catalog allows a CI Type to be designated as a Service, allowing customers to create requests against that service

Dynamic SQL Field Mapper Design & Implementation

- Any FootPrints field can be designated as a Dynamic SQL Lookup Field
- Lookup field can be configured to query any external database
- Data from external source can be configured to map back to FootPrints fields, and automatically populate them when in real-time

Dependent Fields Design & Implementation

- Administrator can configure dropdown field selection in a form to limit the available choices of other dropdown fields in the issue form
- Administrator can configure dropdown field selection in a form to show additional sets of fields in the issue form

Service Level Management Design & Implementation

- Administrator can define service levels with an associated Response Time and Resolution Time
- Response Time and Resolution Time can follow a customized work calendar with holidays and work hours, or a 24-hour clock
- A Service Level is associated with a customer
- Due Date and Response Date values are automatically populated in issues based on a customer's associated service level

Change Management Design & Implementation

- CM Process defines criteria to start the first Phase belonging to that process
- CM Process contains one or more Phases
- CM Phase defines eligible voters and voting criteria for approval and disapproval
- CM Phase defines actions on approval, disapproval or expiration

Part-time:

Entwined Studio (2008-Present)

Cofounder and photographer for wedding photography business that has been featured in The Knot Magazine, Style Me Pretty, New York Magazine's Wedding Edition, W Magazine, Rock'n'Roll Bride, The Bride's Cafe, Essence.com, GetMarried.com, Black Bride and Groom Magazine and in the "Down The Aisle" section of The New York Post

Note: This does not interfere with my software development priorities. I love software development, and want to keep it in my life.

Bridable (2009)

Created a wedding planning community in Drupal that includes forums, friendships, personal blogs, user profiles, articles, videos, and searching for vendors by type within the proximity of a specified (city, state) or zip code. Planned to acquire traffic and then charge vendors for premium listings.

Rutgers (1996-1997)

Computer lab technician, assisted users with software and hardware troubleshooting.

C&G Technologies (1997)

Wrote printer driver software in C for a multi-function scanner/printer/fax machine.

BBS Software (1992-1994)

Wrote, released, and profited from my own Bulletin Board System software at the age of 16. It was a "social networking software" before the word, or even the web, was popular. It was written in C. My blog post about this was front-paged on HackerNews.

Skills & Interests

Front-end:

JavaScript, JavaScript frameworks (Vue, React, jQuery, YUI2 & YUI3, ExtJS), CSS, CSS selection, Bootstrap 3 and 4, Semantic UI, templating, HTML, HTML DOM, AJAX, Photoshop, User Experience & Interaction, prototyping, and conducting user research.

In-between:

JSON, XML, CGI (mod_perl, fastcgi), Cookies, Session Variables, Web Services, REST APIs, SOAP, OAuth.

Back-end:

PHP, Node.js, Perl, C, C++, Python, MVC Frameworks (especially Laravel), Express.js, Django, OO and procedural development methodologies, SQL, Relational Databases (MySQL/MariaDB, Oracle, Postgresql, MSSQL), NoSQL (MongoDB), Elasticsearch, Caching systems (Memcached), Redis, Messaging (RabbitMQ, SQS), performance and database optimization.

Misc:

Security best practices, saas & multitenancy, requirements gathering and writing, logical data modeling, process refinement, creating development guidelines & coding standards, full SDLC, timeline management, capacity planning, leading post-mortems & project retrospectives, Agile, Scrum, version control (Subversion, Git, Hg), Travis CI. I know my way around Apache/IIS/Nginx, *nix, Windows, and Mac OS equally well. Experience using Vagrant, Puppet, Docker, VirtualBox, InstallShield, DigitalOcean and Amazon Web Services (EC2, Lightsail, RDS, SES, EBS, S3, Route53, CloudFront, CloudWatch, Elastic Beanstalk, Elasticsearch Service, ElastiCache).

Education

University:

Rutgers College (1995-99)

- BS in Computer Science
- Minor in Psychology

Certification:

ITIL V3 Foundation in IT Service Management (2008)