

Matthew Martin

Herndon VA, United States
571-247-6010
matthewdeanmartin@gmail.com
<http://github.com/matthewdeanmartin>

General Career Goals

Capable of working as a supervisor or advisor to developers, which covers job titles such as Technical Lead, Team Lead, Project Manager, Application Architect.

Capable of doing server administration and application development work, which covers job titles such as DevOps Engineer, Release Engineer, Build Master.

Experience

Technical Lead and Python Developer – Artemis Consulting (renamed Aretum)

Sept 2019 → now

Tech lead for a python-based search engine for legal documents at Library of Congress, USCO.

Worked with team of specialists on a cloud native application. I mainly wrote python and terraform scripts, but also many others, see below. I helped team members use developer tooling IDEs and build scripts. Set up devcontainers to help with application and tooling installation. Also used AWS Cloud simulation tools such as localstack and python's moto library.

Participated in SAFe-style Agile, including 3 month PI planning, 3 week sprint planning, ticket writing, and following up with each developer on ticket completion to ensure that the application stayed stable, working and demonstrating it each three weeks, all year round for 5 years, with no gaps between sprints. Gathered metrics and helped client interpret metrics to ensure that metrics were fair and helped guide continuous improvement.

Participated in full Federal government style process and SDLC, including A&A, ATO, responding to POAMs, and execution of Change Requests (production releases).

Participated in technical interviewing, developer on-boarding, and dealing with personnel situations including mentoring and PIPs.

Keywords: Python, Connexion/Gunicorn, OpenAPI, ECS, ECR, S3, Glue, RDS, EC2, ALB, API Gateway, ElastiCache, Opensearch/Elasticsearch, Nginx. Integration accomplished with Terraform, Docker, Podman, Gitlab CI. Quality verified with SonarQube, pylint, pytest, locust (load testing), mypy, unit testing, Makefile, Justfile. Incidental scripting in bash on Windows, Mac, Linux and headless AWS Linux. Supervised automated testing with Java NgTest/Selenium. Supervised other developers creating an Angular application, mostly contributing in code review, writing unit tests, and automated quality assurance tools, such as typescript linting, etc. Used a variety of mandated and elective security scanning tools such as safety, tenable, trivy, OWASP ZAP. Configuring applications to use Keycloak, a JWT-based technology. VSCode, JetBrains' products such as WebStorm, IntelliJ, PyCharm

Systems Architect – ASRC

Mar 2019 → Sept 2019

Support contract for USPTO

Worked with Architecture Division to help introduce new technologies through out the organization. Container work using Docker, Kubernetes and Jenkins.

Brief work with Azure, particularly the pre-LLM LUIS chatbot.

Brief work with UIPath's RPA offering and creation of a proof of concept for application to Ops teams.

Brief work deploying Xibo signage/kiosk application to EC2 using a lift-and-shift architecture.

Scrum Master and Senior Development Engineer – Burson Marsteller (renamed Burson Cohn & Wolfe)

Jun 2016 → Sept 2019

python, amazon-web-services, postgresql, docker, ansible, microservices, ec2, sqs, rds, ecs, iam, aws-step-functions, s3, emr, apache-spark, sk-learn, ubuntu, team-city

Responsible for team supervision, writing code, build mastering and server administration for AWS infrastructure.

Managed between one and five developers, supported three code bases, shipped two major versions of our main application. My team cut AWS bills, responded to operational requests. I also did new development including Sk-learn modeling and social media reports.

- Evaluated technology options, did proofs of concept and provided these proofs of concept to developers so they could implement the rest themselves. I used this pattern for our build scripts, dockerfile, ansible scripts, and python projects
- Reworked our server build from manually reconstructing bash history to using Ansible scripts. Briefly tried out Puppet and Vagrant.
- Wrote a comprehensive build script in bash. That script outgrew the limitations of bash as a programming language and I rewrote the script in python.
- Split our monolith application into five components, two microservers using Swagger REST APIs running on Docker. Swagger is a yaml configuration file based tool for exposing ordinary Python as a REST API.
- Did server administration for 30-60 servers on AWS. Mostly ubuntu, sometimes CentOS (RHEL). Tasks included taking inventory of assets, looking for cost saving opportunities, continually hardening the virtual network, migrating machines from less secure VPCs to more secure VPCs, managing users and API keys
- Researched architectural options and migrated application components from bad choices to better choices, such as SQS to Redis, and Celery to AWS Step functions—all of those are types of message queues
- Supervised one DevOps engineer who participated in most of the above tasks. Interfaced with organizational server admin to ensure there was backup continuity for all important tasks.
- Supervised a temporary team of developers. Got team up to speed on development environment, help team troubleshoot setup issues to make sure they could write code that would integrate with existing application.

Senior Software Engineer I – Forsmarsh Group

Jan 2015 → Jun 2016

asp.net, asp.net-web-api, wif, python, asp.net-mvc-4, wcf

I did maintenance development releases of a Python 3 GUI application. The application was for a research study of how people create passwords. The first release needed a few known bugs fixed and demonstrate that no additional bugs were introduced. To achieve this, the legacy application was refactored and put under a large number of automated unit tests.

Once the legacy application was stabilized, I created a new internal library over the TKInter library to simplify creating new screens. The application was instrumented for diagnostics and trace so that issues could be identified by examining logs.

The final application included a build script, machine generated API documentation and a packaged application that could run on a machine without a pre-existing Python runtime.

Tools used PyCharm, Eclipse, Iron Python

Libraries used unittest, nose, sphinx, pywin, virtualenv

.NET Stack Backend Development Created entity framework data tier for authentication store. Augment existing WIF (Windows Identity Framework) single sign on solution to support self configuration. Create HTTP/REST based ASP.NET WebAPI interface and corresponding C#, JavaScript and PHP clients.

Participated in maintenance of both WebForms, MS-Ajax and MVC 4 applications.

Created WCF service layer, including features for reducing configuration and misconfiguration risks. Also did WCF performance tuning, which included improving serialization and switching to asynchronous techniques using .NET 4.0 Task library.

Create data import tool using ASP.NET, jQuery and jQuery-UI to deliver a solution faster and with more quality assurance checks than could have been implemented in WebForms.

HTML5 and TypeScript Front End Development Implemented requirejs module loading, converted existing JavaScript to TypeScript. Implemented clientside error logging, client side random data generation for testing and underscore based widget templates. Participated in creating knockout-js client side databinding for an MVC application.

Created workflow for the periodic import of a complex Drupal-based HTML template from an external organization in a way that the same artifacts could be used to style several ASP.NET webforms applications as well as an MVC application.

DevOps and Buildmastering

I established an end-to-end automated TFS build, customized for NUnit tests. Wrote powershell scripts to automate certain build tasks and deployment tasks.

Senior .NET Developer – Benten Technologies Jul 2014 → Dec 2014

asp.net-web-api, asp.net-mvc, c#-4.0, tfs, entity-framework, wif

Benten was the hiring agency that employed me to work at ForsMarsh until my contract converted to full time employment at

ForsMarsh. The contract and location of employment stayed the same throughout.

JavaScript and C# Software Developer III – CACI Feb 2013 → Jul 2014

javascript, tfs, asp.net, c#, .net-4.0, jquery, wcf, jquery-ui, task-parallel-library

OracleForms to .NET Application Conversion. Worked on a large team of three to six developers in an application re-write.

We migrated the user interface from Oracle Forms to a hybrid ASP.NET WebForms, MVC and JavaScript driven single page application, with some work to integrate with Oracle backend using PL-SQL. Significant time spent improving web application performance to get the web application which started out with 10-15 second response for most action down to instant response time.

.NET Performance Tuning. I also did extensive performance for an application that generated Word documents, which at the beginning could run overnight and often failed anyhow. I solved the application's multi-threading problems, races, deadlocks, implemented Parallel Task Library and WCF patterns to get the reports to generate in three to five minutes, despite having to do 50 times the work load.

DevOps and Buildmastering. Refactored legacy code and implemented automated unit and integration tests to detect breaks before release to test. Participated in the release process using TFS as a process management tool and build server.

Saw about six maintenance releases to production and 50 iterative releases to test as a part of a broader continuous integration strategy.

JavaScript and C# Software Developer III – AboutWeb Aug 2012 → Feb 2013 **asp.net, c#, javascript, tfs, iis7**

AboutWeb was the hiring agency that employed me to work at CACI until my contract converted to full time employment at CACI. The contract and location of employment stayed the same throughout.

Senior Consultant – Procentrix, Inc. May 2008 → Aug 2012

c#, visual-studio, vs2010, vs2008, subversion, team-city, flyspray, ssrs, telerik

Full Stack .NET Development I did maintenance development on a line of business application using C#, TSQL, Javascript,

using Telerik components. I upgraded it from .NET 1.1 to 3.5. I created a comprehensive set of automated integration and

automated unit tests (nUnit) that reduced the number of logged errors from hundreds a day to several per week.

I was lead developer on a three person team that moved about 15 releases through TSA's SDLC release process. Task included identifying opportunities for improving the application, architecting the application to allow for collaboration, dividing certain development tasks and code review of other developer's code. I wrote SDLC documents to support release activities. I also did the maintenance development necessary to get the application to pass C&A (Certification and Authorization).

DevOps. I set up a complete continuous integration system, initially using Visual Source Safe (VSS), Luntbuild, and Flyspray issue tracking and eventually upgrading to Subversion, Collabnet issue tracking and TeamCity. I was responsible for managing the servers and workstations on the developer network, including installing operating systems, DBA tasks for MySQL and MSSQL, such as version upgrades, schema comparison and security hardening.

.NET Application Modernization. I refactored a .NET/C#/LLBGen travel scheduler application to enable the use of a majority of the existing codebase for use by a new department. LLBGen is a ORM, similar to NHibernate. I did another modernization upgrade to take an ASP.NET scheduling and travel application to 2.0 and identified the legacy application's XSS and SQL Injection vulnerabilities.

Data Warehouse I extracted data from a schema lacking foreign keys to a star schema so that it could be loaded into a variety of SQL Server Analysis Services cubes.

SharePoint. I created a proof of concept web part for a Sharepoint management statistics dashboard.

Software Developer III – Dynamics Research Corporation May 2005 → May 2008

asp-classic, asp.net, sql-2000, ssas, ssis, ssms, vs2008

ASP Classic to ASP.NET Migration. While working on MERLIN, an ASP classic application, completed both an upgrade to ASP.NET and a rewrite using ASP.NET, taking full advantage of all features of ASP.NET 2.0, including custom and user controls. Created innovative 508 accessibility interface to existing MERLIN using ASP.NET 2.0 profile features.

ASP.NET and T-SQL Website. I wrote a web based data interface tool for providing an interface between an Oracle scheduling and the MS SQL server database that feeds a simulation model. Interface application includes a significant number

of tools and reports for dealing with common data interface scenarios and exceptions.

ASP Classic Modernization. Upgraded ASP classic application to use CSS and web standards, dramatically reducing size and complexity of HTML files and increasing the ease of maintenance development. All improvement done using CMMI repeatable processes

DTS Support and Migration to SSIS. While working on the MERLIN contract, I provide ongoing support for troubleshooting data interface issues that arise in the monthly update of the MERLIN data warehouse. I have upgraded all data import packages from DTS to SSIS, the successor technology to DTS.

T-SQL Maintenance. I wrote T-SQL to implement new metrics and levels of detail into MERLIN. I refactored T-SQL queries to reduce lines of code by factor of five for Weapon System Availability task.

Microsoft SQL Administration. I automated backup and other administrative scripts for database.

Implemented notification and monitoring to ensure health of the data warehouse.

SharePoint Based Issue Tracker. Installed and configured SharePoint issue tracking system for support of software development, which was used in live production to support a geographically distributed software development team.

Successfully used new issue trackers for coordinating internal communications in a three person distributed team.

SQL Server Analysis Service Project. I wrote prototype Microsoft Analysis Services cubes for enabling high performance business intelligence querying. Created proof of concept for use of MDX for data warehouse querying.

Source Control Migration. I collaborated with administrative staff to repair and stabilize VSS repositories and prevent loss of historical source code. Set up Subversion repository for development team to migrate their source code to.

Created developer oriented documentation. I wrote data dictionary and API documentation in support of maintenance tasks for MERLIN. I wrote comprehensive web-based user documentation for "MOS Refresh" data interface tool.

Systems Developer I – Prince William County Oct 1998 → May 2005

vb6, ms-access, sql-server, database

VB6 WinForms Development, T-SQL Development and MS-Access Reports. I did maintenance on a closed source client-server application for the tax office. Tasks included report writing for mailing tens of thousands of heterogeneous tax bills. I wrote code to interoperate with existing application via the SQL Server through triggers, scheduled tasks, and stored procedures.

I also wrote a large integrated tax billing software application to supplement the county's COTS system using MS Access, XML, MS SQL Server and custom unit tests. This project involved all stages of the software development cycle from soliciting requirements to design, to deployment and training.

DTS and Custom Interfaces. I wrote numerous TSQL procedures, DTS, Ms-Access and other custom import and export packages to support interfaces with external organizations, such as banks and other government agencies.

Data Quality. I wrote a variety of MS-Access cross tabs to find data quality errors in a Visual FoxPro system.