

Genius Machado
(201) 492 8115
geniusmachado@gmail.com | [LinkedIn](#)

Summary:

- Over 7 years of experience as a Python Developer, specializing in backend engineering using Django, FastAPI, and Flask to build secure, scalable microservices and RESTful APIs for domains such as finance, retail, and analytics.
- Proficient in designing asynchronous, event-driven applications using FastAPI and Redis Pub/Sub, enabling high-throughput messaging, real-time dashboards, and distributed locking for mission-critical systems.
- Built cloud-native applications using AWS services including EC2, RDS, Lambda, S3, DynamoDB, Glue, CloudWatch, IAM, KMS, SNS, and SQS, provisioning infrastructure with Terraform, AWS CDK, and CloudFormation across multi-account environments.
- Developed full-stack platforms using React.js, AngularJS, Next.js, TypeScript, and Node.js, implementing dynamic interfaces integrated with RESTful and GraphQL APIs. Delivered interactive visualizations with D3.js, Highcharts, and Recharts.
- Strong experience with SQL and NoSQL databases including PostgreSQL, MySQL, Oracle, MongoDB, and Redis, using Django ORM and SQLAlchemy for schema design, query optimization, and data access layer abstraction.
- Designed and managed ETL pipelines using Apache Airflow, AWS Glue, and Spark, integrating structured and unstructured data from Kafka, Kinesis, and S3 into Snowflake and Redshift for analytical workflows and ML inference.
- Containerized services with Docker and orchestrated them on Kubernetes (EKS, GKE, OpenShift) using Helm, applying autoscaling, service mesh, and rolling deployments for resilient and scalable microservices.
- Built CI/CD pipelines using Jenkins, GitHub Actions, GitLab CI, and SonarQube to automate test, build, and deployment workflows. Applied TDD and BDD with Pytest, Behave, and Selenium for quality enforcement.
- Integrated LLM and GenAI capabilities using OpenAI and AWS Bedrock with Retrieval-Augmented Generation (RAG) for natural language querying, semantic search, and insight generation over Redshift and MongoDB.
- Experienced in implementing secure authentication and authorization using OAuth2, JWT, and LDAP integration, ensuring compliance and multi-tenant access control across enterprise-grade systems.
- Automated infrastructure and operational tasks with Bash and Python scripting for ETL validations, data sync checks, system bootstrapping, and cron-based maintenance across environments.
- Applied Agile methodologies across all roles, actively contributing to sprint ceremonies, backlog grooming, code reviews, and technical documentation using JIRA and Confluence.

Professional Experience:

Mass General Brigham

Software Engineer, API Development

August 2025 – Present

- Develop, maintain, and optimize RESTful APIs using FastAPI to enable secure transfer of pathology data between Legacy Viewer, Epic Beaker, and downstream healthcare systems.
- Design API endpoints for data ingestion, retrieval, and transformation to support clinical workflows and research needs.
- Write and refine SQL queries in Snowflake and PostgreSQL to handle large imaging datasets, ensuring accuracy, efficiency, and scalability.
- Implement stored procedures and query optimizations to reduce response time and improve performance for high-volume data requests.
- Collaborate with cross-functional teams, including pathologists, data scientists, and IT administrators, to align API functionality with clinical requirements.
- Integrate Epic Beaker with multiple legacy laboratory systems to ensure compatibility and seamless data accessibility across platforms.
- Provide technical support for Legacy Viewer interfaces by troubleshooting issues related to data flow, latency, and connectivity.
- Resolve performance bottlenecks to improve reliability, scalability, and uptime of data delivery pipelines.
- Enhance API infrastructure through route refactoring, code modularization, and adherence to secure design principles.
- Contribute to system documentation, knowledge-sharing sessions, and best practice guidelines to support ongoing API development and maintenance.

Fitch Ratings

Python developer

October 2023 – July 2025

- Developed scalable backend services using Python and Django, integrating FastAPI endpoints for internal tools and data service orchestration. Leveraged Django ORM with PostgreSQL for efficient query handling, schema design, and relational data integrity.
- Built frontend dashboards using React.js with Redux and TypeScript, connected to Django and FastAPI APIs, enabling real-time visualization of bond rating models, client analytics, and compliance metrics. Integrated Recharts and D3.js for

financial visualizations.

- Designed and orchestrated batch and streaming ETL pipelines using Apache Airflow, AWS Glue, and Kafka, enabling ingestion from third-party providers and publishing enriched outputs to Redshift, Snowflake, and MongoDB.
- Integrated AWS Glue with S3, RDS, and Glue Catalog to standardize metadata and automate schema discovery. Built transformation logic with Python and deployed workflows with parameterization for dynamic execution.
- Implemented LLM-based GenAI services using OpenAI, Bedrock, and LangChain, enabling internal users to perform semantic search and document summarization on unstructured financial data. Built RAG pipelines powered by Redshift, MongoDB, and vector databases.
- Integrated Quartz engines for real-time fixed income analytics and valuation forecasting into backend workflows. Exposed computation layers through secured REST APIs for internal tools.
- Used Redis for caching frequent queries and metadata responses across Django and FastAPI apps, optimizing latency for rating portal APIs and improving throughput under load.
- Developed and maintained CI/CD pipelines using GitHub Actions, Jenkins, and GitLab CI, embedding Pytest, Behave, and SonarQube to ensure test coverage, linting, and secure container images.
- Deployed applications in containerized environments using Docker and orchestrated on EKS with Helm, implementing pod health checks, autoscaling policies, and secure IAM roles via Terraform and AWS CDK.
- Created dashboards with New Relic, CloudWatch, Prometheus, and Grafana to monitor service health, log ingestion rates, and ETL success rates. Configured alerts and distributed tracing for incident detection and RCA.
- Designed and optimized SQL-based analytics in Snowflake and Redshift for bond issuance data, risk segmentation, and rating history. Applied NoSQL modeling with MongoDB and DynamoDB for unstructured reference datasets.
- Automated validation and transformation tasks using Bash, Python, and cron jobs, integrating with Airflow DAGs for pre-processing daily batch loads and API syncs.
- Collaborated within Agile teams using JIRA and Confluence, participated in backlog grooming, code reviews, and release planning. Mentored junior developers on FastAPI architecture, CI practices, and LLM prompt integration.

American Express

Python developer

January 2023 – September 2023

- Designed and developed backend microservices using Python, FastAPI, and Flask, implementing modular business logic for customer profile management and rewards transactions. Developed secured RESTful APIs with OAuth2 and JWT, enabling seamless integration with frontend systems and third-party vendors.
- Built reusable UI components using React.js, Next.js, and Vue.js, integrating them with backend APIs to support dynamic customer dashboards and financial activity visualization. Implemented D3.js-based interactive components and graphs for actionable insights into spending trends.
- Developed and containerized Python services using Docker and deployed them to staging and QA environments using Kubernetes (EKS) with Helm charts. Configured autoscaling and rolling updates to support resilient, zero-downtime deployments.
- Engineered ETL workflows using Apache Airflow, automating ingestion from AWS S3, data transformation, and load into Snowflake for analytics reporting. Optimized complex SQL queries to improve processing time for downstream business intelligence dashboards.
- Optimized Redis integration for caching session tokens and frequently queried financial metadata, reducing latency and improving load handling. Designed and applied cache invalidation policies to maintain consistency across authentication layers and data APIs.
- Assisted in infrastructure provisioning using Terraform and AWS CDK, automating deployments of EC2, Lambda, S3, and RDS components across dev and QA environments. Defined reusable Terraform modules for version-controlled infrastructure templates.
- Contributed to infrastructure configuration using Ansible, managing repeatable deployments for local dev setups, test environments, and EC2-based microservice provisioning. Assisted in transitioning legacy scripts to standardized Ansible playbooks.
- Participated in building CI/CD pipelines using GitHub Actions and Jenkins, embedding Pytest, Selenium, and SonarQube checks into build workflows to enforce test coverage, code quality, and secure deployments.
- Supported A/B experimentation using feature flags in PostgreSQL and UI toggles in React-based apps, enabling product teams to test marketing messages and onboarding flows in real-time with statistical evaluation via NumPy and SciPy.
- Automated operational tasks using Bash and Python scripting, including ETL validation, data archival, cron-based cache purges, and S3-to-Snowflake data sync checks.
- Monitored system metrics using AWS CloudWatch and New Relic, setting alerts for latency anomalies, memory spikes, and API downtime, supporting timely triage and incident recovery.
- Engaged in Agile development cycles using JIRA and Confluence, contributing to sprint planning, retrospectives, story estimation, and documentation of backend architecture and frontend-data flow models.

UnitedHealth Group

Software Engineer

January 2020 – December 2021

- Designed and developed backend microservices using Python, FastAPI, and Flask, modularizing customer profile management and rewards processing workflows. Built and secured RESTful APIs with OAuth2 and JWT, supporting integrations with mobile apps, partner platforms, and internal dashboards.
- Developed reusable, interactive UI components using React.js, Next.js, and Vue.js, integrating with backend APIs for real-time customer dashboard views and spending breakdowns. Embedded D3.js-powered visualizations to represent financial activity and rewards trends across cohorts.
- Containerized microservices using Docker and deployed to Kubernetes (EKS) environments using Helm, enabling scalable staging environments with health checks, autoscaling policies, and rollout strategies for continuous delivery.
- Engineered ETL workflows using Apache Airflow, orchestrating ingestion of transaction logs from AWS S3, transformation pipelines using Pandas, and batch loading into Snowflake for downstream reporting. Tuned SQL queries for business intelligence consumption.
- Integrated Redis for caching authentication tokens, account balances, and user session data. Implemented cache invalidation strategies aligned with TTL policies and Django-compatible cache layers to improve latency and reduce RDS load.
- Supported infrastructure provisioning using Terraform, AWS CDK, and CloudFormation, automating deployment of Lambda, EC2, RDS, and S3 resources across QA and staging environments. Created reusable Terraform modules for infrastructure templates.
- Configured repeatable local development and QA environments using Ansible, streamlining service onboarding and environment parity. Migrated legacy setup scripts to Ansible playbooks and integrated them into CI workflows.
- Developed CI/CD pipelines with GitHub Actions, Jenkins, and SonarQube, embedding Pytest and Selenium for functional and regression testing. Enforced security and linting standards across microservice deployments.
- Built A/B experimentation framework using PostgreSQL-based feature flags, React toggles, and statistical evaluation powered by NumPy and SciPy, helping product teams assess reward tier effectiveness and personalization models.
- Automated devops and ETL processes using Python and Bash scripts, managing file ingestion validation, cache refresh routines, and cron-based S3-to-Snowflake archive syncs integrated into Airflow DAGs.
- Implemented observability using AWS CloudWatch, New Relic, and custom metrics, enabling distributed tracing, alerting for API latency spikes, and memory anomalies across containerized services.
- Participated in Agile sprints via JIRA and Confluence, contributing to user story grooming, sprint planning, backend documentation, and architecture reviews in a collaborative multi-team setup.

Barclays

Software Developer

July 2018 – December 2019

- Developed and optimized backend services using Python and Django, implementing secure RESTful APIs to support real-time event tracking, transaction auditing, and data aggregation pipelines. Integrated Django ORM with MySQL for efficient query execution and transactional integrity.
- Built data processing workflows and dynamic reporting modules using Python, Django REST Framework, and MySQL, delivering analytical dashboards that visualized live financial metrics through Highcharts and HTML5 canvas rendering.
- Designed responsive frontend components using HTML5, CSS3, and JavaScript, incorporating jQuery for DOM manipulation and asynchronous interactions. Delivered UI enhancements to streamline user workflows and improve cross-browser compatibility.
- Created and integrated custom Highcharts-based dashboards for data visualization, enabling real-time insights into transaction trends and risk indicators. Connected charts to Django APIs for seamless, dynamic rendering based on query filters and user inputs.
- Conducted unit testing and integration testing using Pytest and Django's built-in test framework, validating endpoint correctness, database integrity, and frontend-backend interactions. Implemented QA automation for regression testing of core API services.
- Collaborated in Agile development teams using JIRA for sprint tracking and Confluence for documentation. Participated in sprint planning, code reviews, and bug triage, contributing to performance improvements and deployment stability.
- Refactored legacy Django views and serializers to improve performance and scalability for high-volume data operations. Optimized SQL queries and indexing strategies in MySQL to reduce API response times.
- Deployed Django applications on Linux environments using Apache and WSGI, configured systemd services for process management, and integrated logging for request tracing and error diagnostics.

Technical skills:

- **Programming Languages & Frameworks:** Python, Django, Flask, FastAPI, JavaScript, TypeScript, HTML5, CSS3, Bash, SQL, Shell Scripting
- **Frontend Technologies:** React.js, Next.js, Vue.js, AngularJS, Redux, jQuery, D3.js, Highcharts, HTML Canvas, Responsive

UI Design

- **API Development & Integration:** RESTful APIs, GraphQL, OAuth2, JWT, Django REST Framework, OpenAI API, Bedrock, LangChain
 - **Cloud Platforms & Services:** Amazon Web Services (AWS) – EC2, Lambda, RDS, S3, DynamoDB, Redshift, Glue, Glue Catalog, Kinesis, EKS, IAM, KMS, CloudWatch, SNS, SQS, AWS CDK, CloudFormation
 - **ETL & Data Engineering:** Apache Airflow, AWS Glue, Apache Spark, Pandas, ETL Pipelines (Batch and Streaming), Snowflake, Redshift, Kafka, Kinesis
 - **Databases:** PostgreSQL, MySQL, MongoDB, DynamoDB (Single-table design), Snowflake, Redshift, Redis (caching, Pub/Sub, distributed locking), Oracle
 - **Containerization & Orchestration:** Docker, Kubernetes (EKS), Helm, systemd, WSGI, Linux-based deployment
 - **Infrastructure Automation:** Terraform, AWS CDK, AWS CloudFormation, Ansible (Playbooks, Configuration Management)
 - **CI/CD & DevOps:** Git, GitHub Actions, GitLab CI, Jenkins, SonarQube, CI/CD Pipelines, Version Control (Git, Bitbucket)
 - **Testing & Quality Assurance:** Pytest, Selenium, Behave, Django Test Framework, TDD, BDD, QA Automation
 - **Monitoring & Observability:** AWS CloudWatch, Prometheus, Grafana, New Relic, Distributed Tracing, Log Aggregation
 - **Visualization & Analytics:** Recharts, D3.js, Highcharts, Redshift Analytics, SQL Analytics, Canvas-based Charts
 - **Machine Learning & GenAI:** OpenAI GPT, AWS Bedrock, Retrieval-Augmented Generation (RAG), LangChain, Vector Databases
 - **Agile Tools & Collaboration:** JIRA, Confluence, Sprint Planning, Code Reviews, Agile Ceremonies
-