KRISHNASIS MANDAL

+49-[on-request] | krishnasis@hotmail.com | Berlin, Germany | linkedin.com/in/krishnasis | github.com/krishnasism

EXPERIENCE

Riverty, Berlin, Germany October 2024 - *Present*

Cloud Software Engineer

- Working with Private Networking services on Azure. Working with PrivateLink, Azure DNS, and Azure VPN. Deploying services on Azure using Terraform. For example, Keyvault, Functions, Container Registry. All services are deployed in a private network.
- Bootstrapping the AKS cluster in the Data Platform team from 0 -> 1. Set up private networking, ACR, Keyvault (for external secrets), DNS, and association with App Gateway and AGIC for Ingress. Set up monitoring and alerting with Grafana.
- Set up e2e testing for AKS service and CI/CD pipelines on Azure DevOps. Working with ArgoCD for app deployment following the app-of-apps pattern.
- Setting up K8s services - ArgoCD, Datahub, Kafka, MySQL, Postgres, and Grafana.
- Focusing on improving the Developer Experience and making the overall processes smoother.. Set up devcontainers to improve the developer experience for local development. Writing how-tos, runbooks, and documentation to easily transfer knowledge within the team and to users.
- Writing microservices for data endpoints using Python (FastAPI & Flask), setting up Helm charts for deployment, and deploying through ArgoCD.
- Deployed an internal PyPI registry in the private network to be compatible with Databricks network isolation. Uses Azure App Service to host the application, and Azure Private Link to communicate with the Databricks network. This removed a huge bottleneck where engineers had to install packages manually on the clusters.
- Building full-stack apps to implement solutions for data engineers. Set up testing and deployment pipelines using Databricks Asset Bundles for CI/CD. Wrote boilerplate applications and deployment pipelines to enable data engineers to build their applications.
- Writing Python packages to provide data engineers and analysts with interfaces for common tools for testing, validation, and moving data between storage.
- Working to reduce overall cloud costs. Optimized Log Analytics ingestion to save ~€4k/month.
- Making architecture decisions related to the overall design and development of the Data Platform.
- Set up Great Expectations on Databricks clusters to allow data engineers to set up quality tests. Wrote a Python wrapper library to allow data engineers to transform and push their results to Postgres with one command - which can be used to set up monitoring on Grafana, across all environments.

Flix, Berlin, Germany May 2024 - September 2024

Software Engineer

- Built Customer Accounts for FlixBus India. Started development in May and went live in September. Used Spring Boot, Auth0, AWS SNS, Terraform, and Kubernetes.
- Set up infrastructure, events, and producer for Kafka.
- Set up metrics, monitors, and dashboards on Datadog (provisioned via Terraform) to monitor application health and Auth0 tenant and attack protection logs, and set up Squadcast alerts for on-call engineers.
- Wrote e2e tests using Playwright (Python) to test the user flow.
- Used Python to build data pipelines to fetch data from compliance and CIAM tools and move them into Snowflake. Used Lambda functions to move data from S3 to Snowflake.
- Used data contracts. Generated Confluence page, Soda quality checks, and automated PII tagging on Snowflake.
- Built AI Compliance Helpdesk to answer questions related to compliance using AWS Bedrock, OpenSearch, Python (FastAPI), React

Prestatech GmbH, Berlin, Germany March 2022 - April 2024

Software Engineer

- Created an open-source PDF parsing Python library: *https://github.com/weareprestatech/hotpdf*.
- Distributed systems design using FastAPI, Docker, K8s, Celery, Redis Queue
- Created Generic Payslips Parsing for German and Italian markets by writing rule-based systems in Python and finetuned document parsing and classification model, LayoutLMv2.

- Deployed AI services with FastAPI. Used Kubernetes, Helm, and ArgoCD to manage infrastructure.
- Trained models on Azure Document Intelligence to augment the pre-existing rule-based extraction systems.
- Worked and deployed a real-estate-focused SaaS product using Vue3, Tailwind and FastAPI (Python).
- Added observability to services using Prometheus and connected them to Grafana for monitoring.
- Integrated Yapily into our systems to fetch PSD2 Transactions.
- Optimised code written in Python to bring down total financial document processing time by 2/3rds. Saved money and improved customer satisfaction.
- Built various parsers to process different types of financial statements in Python (Bank Statements, Payslips).
- Migrated various Azure Function Apps to FastAPI. Deployed to K8s.
- Primary engineer of the Databricks services - used PySpark. Processed ~ 1m transactions per month.
- Saved over €100,000 per year by optimizing Databricks service. Also, optimized cluster configurations to bring down development costs by 97%.
- Implemented end-to-end data processing pipeline in Azure Data Factory.
- Created automated CI/CD & testing pipelines for Azure Databricks workloads in Az DevOps.
- Wrote CI/CD scripts to automate deployment using Az DevOps, and infrastructure using Terraform and Kubernetes. Created pipelines to push Python packages to a private PYPI mirror.
- Created a pipeline for automated release versioning of packages and release notes to Confluence in Azure DevOps, using Python and shell scripting.
- Created an internal tool that automatically crawls through the internal git repositories, extracts dependencies in Python code, lists any vulnerabilities in those dependencies, and then summarizes in a Confluence page using Python.
- Implemented multi-processed parsing of big PDF bank statements to reduce parse time by 10x in Python.
- Researched topics about benchmarking functions, automated builds and deployments, and building new PDF parsers, in depth. Participated in code reviews and pair programming sessions, and wrote plenty of docs.
- Built internal tooling dashboards in Vue.js. Backend using FastAPI.

Miles Technologies, Hyderabad, Telangana August 2020 – February 2022

Software Developer

- Worked with ASP.NET(C#), LINQ, SQL Server, JS, Azure, and Gitlab.
- Wrote custom software for clients using the .NET stack.
- Fixed SQL queries by identifying issues such as parameter sniffing, missing indexes, and unoptimized joins. Implemented a temp table approach instead of column value functions to increase performance. Optimized an SQL function to bring execution time from 10 minutes to less than 3 seconds.
- Created DevOps pipelines using Gitlab to build and deploy .NET applications, and scripts for DB migrations.
- Performed security audits for multiple projects and identified and patched vulnerabilities. Patching vulnerable packages stopped an attack on a system, whereas other projects were compromised globally.
- Implemented login systems and MFA systems. Used Twilio to get customer inputs like Confirm/Reject to SMS.
- Developed a full-fledged text editor with features such as autosaving, cursor restore on reload, and full HTML markup support using JavaScript. Also pushed updates to APIs using the .NET stack.
- Received 3 raving fan merits (client appreciation) and two nominations for top performer of the quarter.

Deloitte, Hyderabad, Telangana July 2019 – July 2020

Associate Analyst (Data Engineering)

- Created big data solutions using Hadoop, Spark stack, and scripting in PySpark and Scala. Used technologies such as Hive, Phoenix, and HBase to extract, transform and load the data.
- Created an Alexa Skill for a client using an AWS Lambda function in Python.
- Maintained and created AWS resources for clients.
- Developed a system in Python as the sole developer for one of the largest healthcare providers in the US, which would process their data and store the data in tables compliant with auditing standards by the government. The system developed decreased the time taken to process by 3 times.
- Created a 2FA system for an Alexa application using AWS services – coded in Python. The system connects to Alexa services using proprietary email systems. Used AWS Lambda, Cognito to implement.
- Optimized a PySpark function to reduce health care big data pipeline processing time by 50%