AI Data Pipeline:

# Slide 1: Title Slide

Hello, and thank you for joining me. My name is Tatenda Manyepa, and this presentation is part of my AWS Solutions Architect portfolio. Today, I’ll walk you through a project titled AI-Powered Data Pipeline for Warehouse Demand Forecasting. This solution focuses on applying machine learning and automation to improve inventory decision-making in warehouse environments.

# Slide 2: Project Overview

This project addresses the challenge of inaccurate demand forecasting, which can lead to overstock, stockouts, and operational inefficiencies. I designed and implemented an AI-powered data pipeline that uses Amazon Kinesis, AWS Glue, Amazon SageMaker, and API Gateway to collect, process, and analyse real-time and historical sales data. The result is a smarter system that enables warehouses to predict demand more accurately and optimise inventory levels accordingly.

# Slide 3: Architecture Diagram

Here’s the high-level architecture that powers the solution. It’s built around a modular, event-driven pipeline: Kinesis Firehose captures real-time inventory and sales data. That data is stored in Amazon S3, and transformed via AWS Glue. Lambda is used to trigger the ETL jobs. SageMaker handles model training and deployment. And API Gateway, integrated with Lambda, serves the real-time predictions. Everything is orchestrated securely within a VPC, and automated with EventBridge, while CloudWatch and CloudTrail provide monitoring and auditing.

# Slide 4: Architecture Flow Summary

Here’s a breakdown of the full flow across five main components: 1. Data Ingestion: Kinesis Firehose ingests real-time warehouse data and stores it in S3. 2. Data Processing: AWS Glue transforms and cleans the data, which is then saved in a structured format. 3. Model Training: SageMaker trains a forecasting model and deploys it as an endpoint. 4. Real-Time Predictions: API Gateway receives forecast requests, which Lambda routes to the SageMaker endpoint. 5. Automation & Monitoring: EventBridge triggers retraining, and CloudWatch and CloudTrail provide full visibility and governance.

# Slide 5: Key Implementation Steps

To bring the pipeline to life, I followed these key implementation steps: Set up Kinesis Firehose and S3 to collect raw data. Created AWS Glue ETL jobs triggered by Lambda. Trained a machine learning model using SageMaker, and stored the model artifacts. Set up API Gateway and Lambda for external predictions. Automated retraining with EventBridge. Secured everything with IAM, KMS, and private subnets. Finally, I validated the results and fine-tuned hyperparameters.

# Slide 6: Challenges & Resolutions

Like any real-world project, I faced a few technical challenges: I adjusted Kinesis Firehose buffer settings to reduce ingestion latency. Resolved Glue job failures by correcting IAM permissions and handling schema mismatches. Optimised SageMaker training time and cost using Spot Instances and better preprocessing. Tackled API Gateway timeouts by extending Lambda timeouts and optimising payloads. And fixed EventBridge triggers by adjusting permissions and validating the cron expressions.

# Slide 7: Reusability & Integration

This pipeline isn’t just for warehouse forecasting. It’s designed to be modular and reusable, meaning it can be repurposed across many industries: Retail for seasonal demand forecasts, Finance for fraud detection, Logistics for supply chain optimisation, and Healthcare for resource planning. It’s also fully compatible with AWS-native services like Lambda, Glue, SageMaker, and API Gateway.

# Slide 8: Results and Impact

The results were significant: I achieved a 25% improvement in forecast accuracy, helping reduce overstock and stockouts. ETL processing times were reduced by 40%. By leveraging Spot Instances and tuning models, I cut ML training costs by 30%. The pipeline is automated, scalable, and delivers reliable real-time forecasts.

# Slide 9: Expected Benefits

Looking at the long-term value, this solution provides: Optimised inventory management through more accurate forecasting. Cost savings and resource efficiency by using scalable cloud-native tools. Real-time business intelligence via on-demand API forecasts. Automation with self-learning models, thanks to EventBridge retraining. And because it’s industry-agnostic, it can be adapted to any environment where demand forecasting matters.

# Slide 10: Thank You and Closing Slide

Thank you for taking the time to explore this project. This AI-powered pipeline is part of my mission to design secure, scalable, and intelligent cloud-native solutions using AWS. You can find more projects on my portfolio at cloudportfolio.co.uk or reach out directly. I look forward to connecting, learning, and innovating together.