CDW Documentation

Operational Plan: Managed AI Services Team

Operational Plan: Managed AI Services Team

1. Team Structure & Roles

Role	Responsibilities
AI Platform Engineer	Deploy, monitor, and manage LLMs on cloud and on-prem infrastructure
DevOps Engineer	Automation scripts, CI/CD for model updates, shell-level ops, infrastructure
Python Developer	Code AI workflows, model wrappers, APIs, data pipelines
Cloud Ops Specialist	Platform-specific expertise (Azure AI, SageMaker, Vertex AI)
Monitoring & Incident Lead	Alerting systems, root cause analysis, remediation workflows
Customer Success Engineer	Handle service requests, documentation, usage monitoring

2. Scope of Services

AI Platforms Only:

- Azure: Azure OpenAI, Azure ML, Cognitive Services

- AWS: SageMaker, Bedrock, AI Services (Comprehend, Rekognition, etc.)

- Google: Vertex AI, PaLM, Generative AI Studio

On-Prem LLMs:

- Models like LLaMA, Mistral, GPT-J hosted on Linux VMs with GPU

Support Domains:

- Shell access, CLI tools (AWS CLI, gcloud, az)

- Python scripting and notebooks

- Automation (Bash, Terraform, Python)

- Monitoring (Prometheus, Grafana, Azure Monitor, CloudWatch, Stackdriver)

- Issue response and remediation

3. Core Operations

Provisioning & Deployment

· Use IaC tools (Terraform, Bicep, Deployment Manager)

· Maintain VM templates with CUDA, PyTorch, HF Transformers

· Bootstrap scripts for API/endpoint setup

Automation & Shell Command Support

· Secure shell (SSH) access with audit logging

· Model lifecycle scripts (start, stop, etc.)

· CI/CD pipelines for model deployment

Monitoring & Observability

· System monitoring: CPU, GPU, disk

· Model monitoring: latency, error rate

· Platform monitoring: Azure Monitor, CloudWatch, Stackdriver

· Alerts via Slack/Teams/email with PagerDuty/Opsgenie

Python Programming Services

· Support JupyterHub

· Maintain Python utility libraries (logging, retry, chaining)

· Support SDKs: openai, boto3, google-cloud-aiplatform, transformers

Issue Remediation Workflow

· Detection – Alert received

· Classification – Severity assessment

· Investigation – Logs, shell, diagnostics

· Remediation – Patch/redeploy

· Postmortem – RCA documentation

4. Security and Access Control

· RBAC and IAM per platform with least privilege

· Bastion/JIT SSH for VM access

· Audit logs on shell, API, model usage

· Data encryption at rest and in transit

5. Toolchain

IaC: Terraform, Bicep, Deployment Manager

Monitoring: Prometheus, Grafana, ELK, Azure Monitor, CloudWatch, Stackdriver

Automation: GitHub Actions, Azure DevOps, Lambda, Cloud Functions

LLM Frameworks: HuggingFace, LangChain, OpenAI SDKs

Issue Mgmt: Jira, Confluence, PagerDuty/Opsgenie

CI/CD: GitHub, GitLab, Azure DevOps

6. SLA & Reporting

Metric	Target
Uptime per LLM Endpoint	≥ 99.5%
Incident Response Time	P1: 15 min, P2: 1 hour
Model Deployment SLA	≤ 4 hours from request
Weekly Reports	Usage, performance, incidents
Monthly Review	Cost, optimization, usage trends

7. Knowledge Management

· Maintain runbooks and playbooks

· Central wiki/documentation

· Training tracks for LLM ops and automation