IMAGINARYCLOUD

Azure Machine Learning Deployment CHECKLIST

1. Environment Preparation

- [ ] Set up Azure ML Workspace with proper RBAC controls
- [ ] Create Compute Targets (CPU/GPU, AKS, ACI, or Batch endpoints)
- [ ] Link required storage (Azure Data Lake, Blob, or SQL DBs)
- [ ] Define virtual network and private endpoints (if needed)
- [ ] Register your environments using Conda or Docker

2. Model Development and Validation

- [ ] Prepare and version your dataset using Dataset class

- [ ] Create reusable pipelines for data prep, training and evaluation
- [ ] Implement hyperparameter tuning using HyperDrive or AutoML
- [ ] Track experiments with mlflow or Azure ML experiment tracking
- [ ] Validate model accuracy, bias, and fairness metrics

3. Model Registration

- [ ] Register final model in Azure ML Registry
- [ ] Attach model metadata: version, author, compliance tags
- [ ] Confirm lineage is tracked (datasets, scripts, environments)
- [ ] Assign promotion stage (dev → staging → production)

4. Deployment Setup

- [ ] Select deployment type (real-time: AKS / ACI or batch endpoint)

- [ ] Create inference configuration (score.py, entry_script)
- [ ] Build an image or environment for deployment
- [ ] Deploy with traffic-splitting if running A/B tests or blue-green
- [ ] Enable logging and monitoring for endpoint

5. Monitoring and Management

- [ ] Enable metrics collection with Azure Monitor and App Insights
- [ ] Configure alerts (latency, failure rate, drift thresholds)
- [ ] Review model outputs for anomalies and errors
- [ ] Set up triggers for automated retraining pipelines
- [ ] Log usage and access events for auditability

6. Governance and Compliance

- [ ] Restrict access via RBAC or Private Link
- [ ] Enable audit logs for model lifecycle activities
- [ ] Mask PII in data pipelines
- [ ] Document promotion criteria and deployment approvals
- [ ] Align with industry/regional compliance standards (e.g. ICO, HIPAA)

7. CI/CD Integration (MLOps)

- [ ] Use Azure DevOps or GitHub Actions for pipeline orchestration
- [ ] Validate models in staging before production push
- [ ] Test rollback workflows and endpoint failover plans
- [ ] Store CI/CD logs, build history, and approvals
- [ ] Version control all assets: code, models, data references