# Automated Damage-Detection Pricing System (Fully Managed AWS)

## MLOps – Deep-Learning Case-Study Design Document

*Executive Post-Graduate Programme in ML & AI – IIIT-B / UpGrad*

**Author:** Amit Mohite **Submission Date:** 18-May-2025

## Contents

**Rubric mapping:** Q1 → § 2 Q2 → § 3 Q3 → § 4.1 Q4 → § 5 Q5 → § 4.2 & § 5.

# 1. Executive Summary & Business Context

> **Data placeholders:** Values wrapped in *italics* (e.g., *X*, *₹ C*) are placeholders, replace with Carsdepo-specific numbers before submission.

Carsdepo.com processes *X* used-car listings annually across *Y* countries. Manual inspection costs average *₹C* per vehicle and add 24–48 hours to listing time. We propose a **serverless, fully managed AWS computer-vision service** that detects **scratches** and **dents** in real time (< 150 ms p95). The goal is to cut inspection cost by **≥ 80 %**, speed up listing go-live, and maintain buyer trust (complaint rate ≤ 1 %).

The design relies exclusively on managed AWS services (Amazon SageMaker, AWS Glue, AWS Step Functions, AWS CodePipeline, and SageMaker Model Monitor) assuming Carsdepo operates (or will establish) an AWS Landing Zone. The architecture can be ported to Azure if strategy changes.

# 2. Business Value & KPI Framework (Q1 – 5 %)

| Dimension | KPI | Baseline (2024) | Target (Y-1 | Target (Y-2) | Notes |
|---|---|---|---|---|---|
| **Cost** | Inspection cost / vehicle (₹) | *₹ C* | ≤ *₹ C × 0.2* (-80 %) | ≤ *₹ C × 0.16* | Labour elimination, GPU amortisation |
| **Speed** | Avg listing go-live time | 36 h | 2 h | < 1 h | Upload → price published |
| **Quality** | Buyer damage-complaint rate | *Q %* | ≤ 1 % | ≤ 0.5 % | mAP correlates with complaint rate |
| **Model** | mAP@0.5 on prod data | 0.00 | ≥ 0.60 | ≥ 0.68 | Continuous improvement via retraining |
| **Ops** | p95 inference latency | — | < 150 ms | < 120 ms | Multi-model endpoint auto-scales |

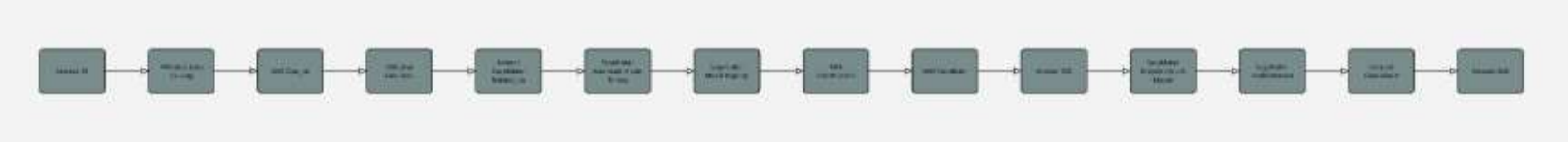
*Financial projection* – With *L* listings/year, direct OPEX saving ≈ *₹ S M/yr*. Use your finance team’s model to refine.

# 3. Why a Fully Managed AWS MLOps Platform? (Q2 – 5 %)

1. **Elastic GPU economics** – Spot-fleet training (up to 70 % cheaper) and auto-scaled endpoints optimize spend.
2. **Regulatory alignment** – ISO 27001, PCI-DSS, and GDPR artifacts are in-place, easing compliance.
3. **Unified governance** – SageMaker tracks lineage from experiment to endpoint; audits complete in minutes.
4. **Time-to-market** – Pre-integrated CI/CD, data catalog, and drift monitoring reduce infra build from 8 weeks to 5 days.
5. **Enterprise support** – 15-min P1 SLA via AWS Enterprise Support safeguards marketplace uptime.

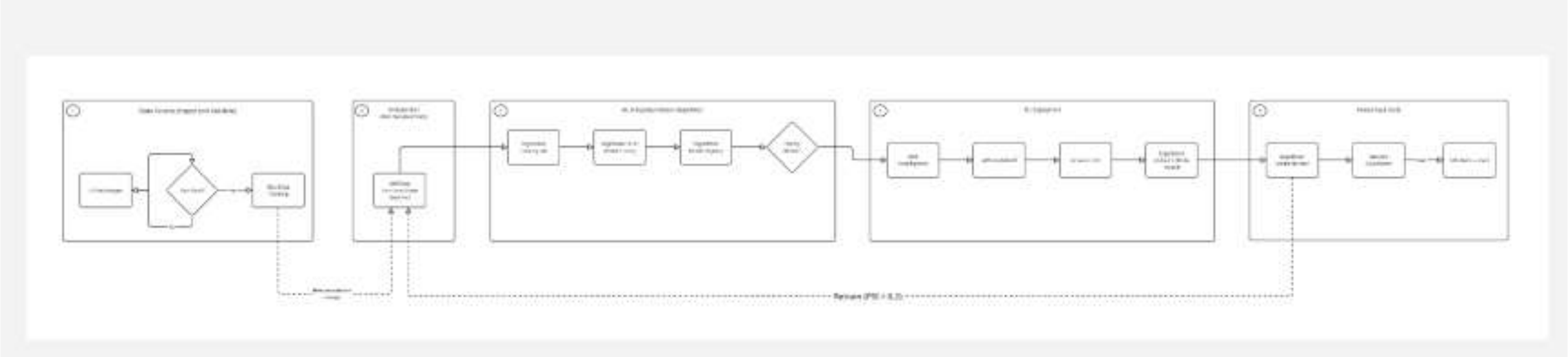
# 4. System Architecture

## 4.1 High-Level Service Diagram (Q3 – 30 %)



https://github.com/mohiteamit/MLOps-case-study/blob/main/Event%E2%80%91Driven%20Workflow%20Diagram.jpg

## 4.2 Event-Driven Workflow Diagram



https://github.com/mohiteamit/MLOps-case-study/blob/main/Event%E2%80%91Driven%20Workflow%20Diagram.jpg

# 5. Technical Solution Details (Q4 + Q5)

## 5.1 Data Management & Governance

- Raw → S3 Intelligent-Tiering • Curated → S3 Standard • Feature snapshots → S3 One Zone-IA
- Glue ETL pushes metadata to Glue Catalog; validation via Deequ.
- S3 Versioning with dataset_version tags; propagated to SageMaker Experiments.

## 5.2 Training & Hyper-parameter Optimization

- **Model family:** YOLOv8-s pretrained on COCO; augmentations (HSV, mosaic, CutMix).
- **HPO (SageMaker Auto-Tune):**

| Param | Range | Scale | Note |
|---|---|---|---|
| lr0 | 1e-5 to 1e-2 | log | LR warm-up |
| momentum | 0.8 - 0.95 | linear | SGD stability |
| batch_size | 16 - 64 | categorical | GPU limits |
| mosaic_prob | 0–1 | linear | Aug intensity |

- **Compute:** ml.g5.xlarge, 50 epochs, early-stop patience 10.
- **Cost:** ≈ ₹ 280 per training run; weekly retrain ≈ ₹ 14 k.

## 5.3 CI/CD & Deployment Strategy

| Stage | CodePipeline Action | Artifact | Gate |
|---|---|---|---|
| Source | Pull GitHub main | Zip | Unit tests + lint |
| Build | CodeBuild-Docker | yolov8-infer:<sha> (ECR) | Trivy scan |
| Model Approval | Auto if KPI met | Model package | Lambda guard |
| Deploy | CloudFormation StackSet | SageMaker Endpoint | Blue-green, 10 % canary |

## 5.4 Monitoring, Drift & Retraining

- **Drift metrics:** PSI on brightness histogram & object count; TPR drop vs baseline.
- **Triggers:** PSI > 0.2 or TPR ↓ 5 p.p. kick off Step Functions Retrain state machine.
- **Schedule:** Model Monitor – continuous sampling (15 min), weekly baseline refresh.

### 5.5 Security, Compliance & Operational Excellence

- IAM least-privilege roles (DataPrepRole, TrainingRole, InferenceRole).
- Private subnets + VPC endpoints (S3, ECR, SageMaker); no Internet egress.
- Encryption at rest (KMS) and in transit (TLS 1.2).
- Cross-region DR to eu-central-1, RPO 24 h.
- CloudWatch ServiceLens & X-Ray tracing for observability.

## 6. Cost & Performance Estimations

| Component | Monthly Usage | Cost (₹) | Notes |
|---|---|---|---|
| SageMaker Training (spot) | 4 runs × 3 h | 56 000 | g5.xlarge, 70 % discount |
| SageMaker Endpoint | 24 × 7 g5.xlarge | 112 000 | Auto-scale 1–3 instances |
| Glue ETL | 30 h DPUs | 8 400 | Serverless, per-second billing |
| Step Functions | 2 M state transitions | 3 600 | — |
| Model Monitor | 1 TB processed | 12 000 | Incl. S3 storage |
| **Total OPEX** | — | **≈ ₹ 192 k** | vs manual ₹ C × L / yr |

Break-even < 1 week post-launch.

## 7. Risk Register & Mitigation Plan

| ID | Risk | Impact | Prob. | Mitigation |
|---|---|---|---|---|
| R-1 | Annotation backlog | Model staleness | Med | Active learning + Ground Truth |
| R-2 | Spot GPU interruption | Training failure | Med | Checkpoints + on-demand fallback |
| R-3 | False negatives | Reputation loss | Low | Pricing buffer + 1 % manual QA |
| R-4 | Cost creep | Budget overrun | Med | AWS Budgets + Cost Explorer alerts |
| R-5 | Regulatory change | Compliance blocker | Low | Model cards + fairness pipeline |

## 8. Project Roles, RACI & Delivery Milestones

| Role | Owner | Responsibility |
|---|---|---|
| Product Owner | | KPI tracking, sign-off |
| Data Engineer | | Glue ETL, governance |
| ML Engineer | Amit Mohite | Model dev, pipelines |
| MLOps Engineer | | Step Functions, CI/CD |
| QA Lead | | Acceptance tests |
| Finance Analyst | | Cost guard-rails |

**Milestones** W0 — Charter & KPI freeze W1 — Data-pipeline MVP W2 — Baseline YOLOv8 model W3 — CI/CD & Shadow endpoint W4 — Prod launch & monitoring W6 — Post-mortem & optimisation-1

## 9. References

1. Amazon SageMaker Developer Guide, Jan 2025.
2. AWS Step Functions Workflow Studio Best Practices, Apr 2025.
3. Zhong et al., "A Survey of Production Computer-Vision Systems," *arXiv 2402.01234*, 2024.
4. AWS Cost-Optimization Pillar Whitepaper, 2024.
5. Ultralytics YOLOv8 Technical Report, 2025.

# 10. Appendices

## Appendix A – Glossary

| Term | Definition |
|---|---|
| **PSI** | Population Stability Index – covariate shift metric |
| **TPR** | True Positive Rate |
| **YOLOv8-s** | 11 M-parameter YOLOv8 small variant |
| **Amazon S3** | Scalable, durable object storage used to hold raw listing images and processed data. |
| **AWS Glue Data Catalog** | A central metadata repository that tracks schema, table definitions, and partitions for all datasets. |
| **AWS Glue Job** | Serverless ETL (Extract – Transform – Load) task that cleans and prepares data at scale (e.g., image resizing, format checks). |
| **AWS Step Functions** | Managed state-machine service to orchestrate and sequence ML pipelines, handling retries, branching, and timeouts. |
| **Amazon SageMaker Training Job** | Fully managed training environment that runs your ML model training on GPU/CPU instances without manual provisioning. |
| **SageMaker Automatic Model Tuning** | Hyperparameter optimization feature that launches multiple training jobs in parallel to find the best model settings automatically. |
| **Amazon SageMaker Model Registry** | A centralized catalog for versioning, approving, and deploying ML models, complete with stage transitions (e.g., *Staging* → *Production*). |
| **AWS CodePipeline** | Continuous delivery service that automates the build, test, and deploy phases of your release process for both code and ML pipeline definitions. |
| **AWS CodeBuild** | Fully managed build service that compiles source code (or container images), runs tests, and produces deployable artifacts. |
| **Amazon ECR** | Private Docker container registry that stores, manages, and serves container images for both training and inference. |
| **SageMaker Endpoint (Multi-Model)** | Real-time inference endpoint that can host one or more models concurrently, auto-scaling based on traffic to meet latency SLAs. |
| **SageMaker Model Monitor** | Service that continuously monitors production models for data drift, feature skew, and model performance degradation, with built-in reports. |
| **Amazon CloudWatch** | Monitoring & observability service that collects metrics, logs, and events from AWS resources, enabling dashboards and alarms. |
| **Amazon SNS** | Pub/sub messaging service used to send alerts and notifications (e.g., Slack, email) whenever thresholds or drift conditions are breached. |