

KIM DAJEONG — Project Portfolio

Cloud / DevOps Engineer

PROJECT 1 — HOSPITAL EMR SYSTEM (HYBRID MULTI-CLOUD, IN PROGRESS)

AWS Infrastructure for Hybrid Multi-Cloud EMR — High Availability, Security, ISMS-P

AWS Infrastructure Role · Team Lead · Apr 28, 2026 – Present · 3-person team · 40% contribution

Led team setup: authored shared documentation (Terraform resource naming conventions, WBS, REST API specification, project account configuration) and designed full architecture diagram through team review sessions

Provisioned AWS Bastion Host via Terraform IaC connecting local environment to AWS RDS

Configured ECS, ALB, and ASG for scalable web serving, and set up CI/CD pipeline for automated web image delivery

Managed team schedule, tracked issues as team lead, responsible for end-to-end AWS environment delivery within a hybrid multi-cloud architecture (AWS + GCP + on-premise)

Stack: AWS (VPC, ALB, ASG, ECS, EC2, RDS PostgreSQL, S3, VPC Endpoint) · Terraform · GitHub Actions

PROJECT 2 — ON-PREMISE KUBERNETES INFRASTRUCTURE (3-TIER, HA, CI/CD, OBSERVABILITY)

Full On-Premise Infrastructure — Kubernetes, High Availability, CI/CD, Observability

Cloud/DevOps Engineer Role · Mar 13 – Mar 25, 2026 (13 days) · Solo

Deployed and managed production Kubernetes cluster (1 master, 2 workers) on VMware on-premise environment, installed Prometheus + Grafana + Loki monitoring stack via Helm, with NFS as centralized Pod log storage

Implemented full HA: HAProxy + Keepalived (entry-point) + MySQL Orchestrator (automatic master failover), Bastion Host enforces strict subnet segmentation between external and internal networks

Built a GitOps CI/CD pipeline using GitHub Actions Self-hosted Runner on the internal network, enabling automatic K8s deployment on code changes

Nginx Ingress handles external traffic routing with admin page restricted to internal network only

ProxySQL Read/Write splitting + query caching maintains read service availability during DB failures, K8s CronJob automates NFS backup for data protection and recovery readiness

Stack: Kubernetes · Helm · VMware · HAProxy · Keepalived · ProxySQL · MySQL Orchestrator · GitHub Actions (Self-hosted) · Docker · Prometheus · Grafana · Loki · Alertmanager · NFS

PROJECT 3 — SERVERLESS LOCATION-BASED WEATHER APP (AWS LAMBDA)

Location-Based Weather Visualization via AWS Serverless Architecture

Frontend & Cloud Integration Role · Jan 29 – Jan 30, 2026 (2 days) · 4-person team · 30% contribution

Connected API Gateway-exposed Lambda endpoint to web frontend; implemented JavaScript (Fetch API) async logic to retrieve weather JSON from S3 and bind data to HTML elements based on location

Gained hands-on understanding of the S3 → Lambda → Web serverless data flow

Stack: AWS Lambda · Amazon S3 · API Gateway · HTML5 · JavaScript (Fetch API) · GitHub Actions

PROJECT 4 — ENTERPRISE RPA BUSINESS PROCESS AUTOMATION

Full Enterprise RPA Project — MS Power Automate, ERP/SAP/MES Integration, Process Automation

RPA Developer Role · Apr 2024 – Jul 2025 (15 months) · Team of 3 · Contribution 70%

Designed and developed 12 out of 44 RPA workflows (Phase 1: 6, Phase 2: 6) using MS Power Automate, covering end-to-end from requirements gathering to go-live

Built RPA integration environment connecting ERP, SAP, MES, in-house security programs, and Excel-based business systems

- Phase 1 (Apr – Nov 2024): Automated simple repetitive processes — approx. 1,169 hrs/year saved

- Phase 2 (Apr – Jul 2025): Automated complex business processes — approx. 2,904 hrs/year saved company-wide

Led Phase 2 requirements analysis: conducted company-wide survey to identify automation candidates, defined development priorities, and established project schedule

Stack: Microsoft Power Automate (Desktop/Cloud) · ERP · SAP · MES · Microsoft Excel · Windows Server · Microsoft 365