

Configuration Manual

MSc Research Project
Cloud Computing

Rohit Korlahalli
Student ID: 23303395

School of Computing
National College of Ireland

Supervisor: Sean Heeney

National College of Ireland
Project Submission Sheet
School of Computing



Student Name:	Rohit Korlahalli
Student ID:	23303395
Programme:	M.Sc Cloud Computing
Year:	2025-26
Module:	MSc Research Project
Supervisor:	Sean Heeney
Submission Due Date:	11/12/2025
Project Title:	Configuration Manual
Word Count:	697
Page Count:	8

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Rohit Korlahalli
Date:	11/12/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Rohit Korlahalli
23303395

1 Introduction

This configuration guide offers a full set of instructions on deploying, configuring, and validating the AIOps Ensemble Anomaly Detection Framework on a Kubernetes cluster. The system combines Prometheus-based telemetry operator, FastAPI-driven ensemble anomaly detectives, Pushgateway to ingest data in metrics and Grafana dashboards to visualise and Slack to alert it operates. Additional features are optional features, such as using an Ollama-based DevOps Chat Assistant UI.

The documentation covers all the prerequisite, software requirements, deployment procedures, environment set up procedures, and validation procedures required to replicate the operational environment that has been used in this project.

2 Prerequisites

2.1 Software Requirements

The tools listed below must be installed on the workstation of the one who intends to deploy for effective communication with the Kubernetes cluster -

- **Google Cloud SDK (gcloud CLI)** - for Kubernetes cluster provisioning and KubeAPI server authentication
- **kubectl** - Kubernetes command line utility to interact with the cluster and provision Kubernetes resources like pods , deployments, and services
- **Docker** - Container runtime used for building all of the microservices images and running them on the cluster
- **Python 3.10+** - for API development (FastAPI), and model training and evaluation
- **Nodejs 18+** - for the frontend of Ollama Devops assistant (optional)
- **Helm3** - optional for Prometheus / Grafana helm charts deployments.
- **Git** - for repository checkout

2.2 Cloud Requirements

- **Google Cloud Project** with billing enabled. (Or any other cloud provider of your choice with a managed Kubernetes service)
- **Google Kubernetes Engine (GKE)** cluster : Recommended 3 nodes (e2-standard-2 or higher)
- Container Registry (**GCR or DockerHub**) for hosting images

2.3 Kubernetes Requirements

- Namespace : like **app** or **aiops**
- Role based access (**RBAC**) to deploy Kubernetes resources like Deployments, Pod, Service, and Ingress

3 Repository and Environment Setup

3.1 Clone the Repository

```
1 git clone https://github.com/rokorlahalli/research-project-23303395.git
2 cd research-project-23303395
```

Repository Structure -

```
1 devops-assistant-frontend/ # Optional for Devops Assistant
2 ensemble-fastapi/ #Ensemble API code
3 kubernetes-manifests/ #Kubernetes resources
4 model-training/ #Model training files
5 prometheus/ #Prometheus helm vaules file
6 README.md
```

3.2 Build Docker Image

3.2.1 Ensemble API

```
1 cd ensemble-fastapi
2 sudo docker build -t <registry>/ensemble-api:latest .
```

3.2.2 Ollama Frontend UI (optional)

```
1 cd devops-assistant-frontend
2 sudo docker build -t <registry>/devops-assistant-frontend:latest .
```

3.2.3 Push Images to Registry

```
1 sudo docker push <registry>/ensemble-api:latest
2 sudo docker push <registry>/devops-assistant-frontend:latest
```

3.2.4 Create Kubernetes Namespace

```
1 kubectl create namespace aiops
```

4 Deployment Steps

4.1 Deploy Prometheus and Grafana

Deploy using Helm charts -

```
1 helm install kps prometheus-community/kube-prometheus-stack -n aiops
```

Verify:

```
1 kubectl get pods -n aiops | grep prometheus
```

4.2 Deploy Pushgateway

```
1 helm install pushgateway prometheus-community/prometheus-pushgateway \
2 --namespace aiops
```

Verify:

```
1 kubectl get svc -n aiops | grep pushgateway
```

4.3 Deploy Ensemble API

4.3.1 Create Secret for Slack Webhook

```
1 kubectl create secret generic slack-secret \
2 --from-literal=SLACK_WEBHOOK_URL="<your-webhook>" \
3 -n aiops
```

4.3.2 Apply Deployment and Service

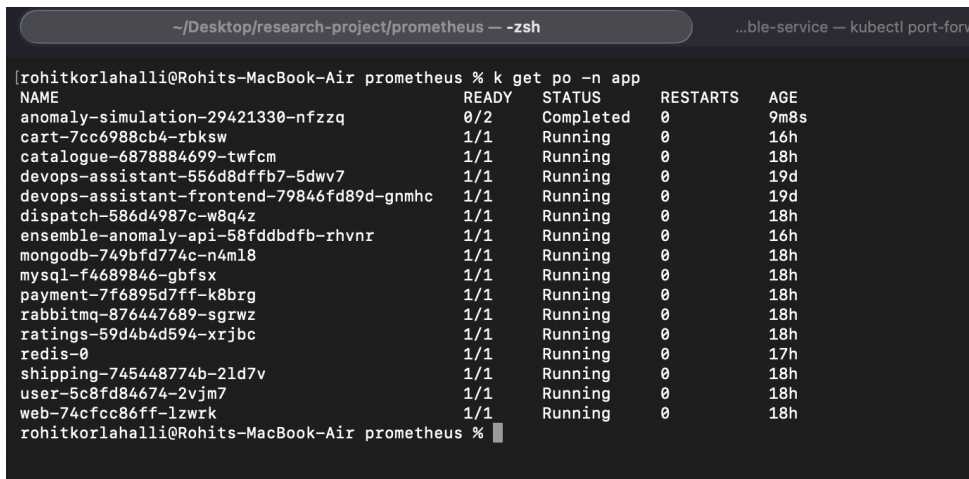
```
kubectl apply -f kubernetes-manifests/ensemble-service/. -n aiops
```

Check deployment health:

```
kubectl get pods -n aiops | grep ensemble
```

4.4 Deploy Ollama Devops Assistant

```
kubectl apply -f kubernetes-manifests/ollama-frontend/. -n aiops
```



NAME	READY	STATUS	RESTARTS	AGE
anomaly-simulation-29421330-nfzzq	0/2	Completed	0	9m8s
cart-7cc6988cb4-rbksw	1/1	Running	0	16h
catalogue-6878884699-twfc	1/1	Running	0	18h
devops-assistant-556d8dff7-5dwv7	1/1	Running	0	19d
devops-assistant-frontend-79846fd89d-gnmhc	1/1	Running	0	19d
dispatch-586d4987c-w8q4z	1/1	Running	0	18h
ensemble-anomaly-api-58fddbdfb-rhvn	1/1	Running	0	16h
mongodb-749bfd774c-n4ml8	1/1	Running	0	18h
mysql-f4689846-gbfsx	1/1	Running	0	18h
payment-7f6895d7ff-k8brg	1/1	Running	0	18h
rabbitmq-876447689-sgrwz	1/1	Running	0	18h
ratings-59d4b4d594-xrjbc	1/1	Running	0	18h
redis-0	1/1	Running	0	17h
shipping-745448774b-2ld7v	1/1	Running	0	18h
user-5c8fd84674-2vjm7	1/1	Running	0	18h
web-74cfcc86ff-lzwrk	1/1	Running	0	18h

Figure 1: Pods Deployed on Kubernetes Cluster

5 Configuration Details

5.1 Ensemble Model Configuration

Default thresholds:

```
ANOMALY_COOLDOWN = 30 seconds
MAJORITY_VOTE = 2/3 models
LSTM_THRESHOLD = 0.5
```

These are stored inside:

```
ensemble-api/main.py
```

5.2 Pushgateway Integration

The API pushes results using:

```
1 POST http://pushgateway.aiops.svc:9091/metrics/job/anomaly_detector
```

Metrics exposed:

- ensemble_anomaly_label
- model_confidence_score
- recommended_action

5.3 Slack Notification Rules

Slack messages include:

- anomaly type
- metric summary
- recommended action
- confidence score

6 Validation and Testing

6.1 Test Ensemble API

```
1 curl -X POST "http://<external-ip>/predict" \  
2 -H "Content-Type: application/json" \  
3 -d '{ "cpu_usage": 0.92, "memory_usage": 0.75, "net_receive": 1200, ... }'
```

Valid Response:

```
1 {  
2   "anomaly": 1,  
3   "model_votes": {"iforest":1, "xgboost":1, "lstm":1},  
4   "insight": "High CPU saturation detected { consider VPA or scaling out."  
5 }
```

6.2 Validate Pushgateway metrics

```
1 curl http://<pushgateway-ip>:9091/metrics | grep ensemble
```

6.3 Validate Grafana Dashboard

- Panels should show time series anomaly labels
- Color overlays mark anomaly intervals
- Statistical trends should be seen under "Cluster Health Overview"

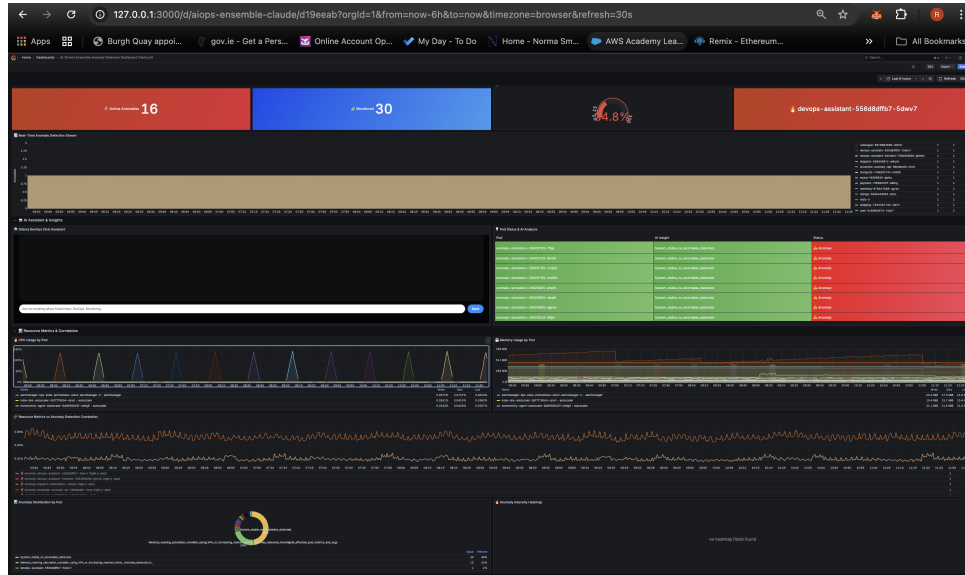


Figure 2: Grafana Dashboard visualizing insights

6.4 Validate Slack Alerts

Send synthetic anomaly:

```
kubectrl exec -it <pod> -- stress-ng --cpu 4 --timeout 20s
```

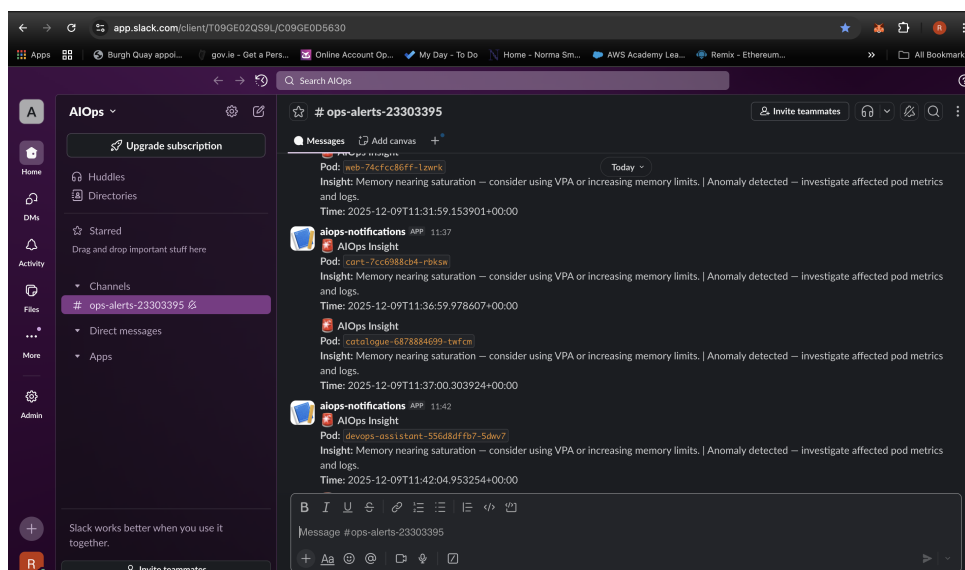


Figure 3: Alerts Received on Slack Channel

7 Common Issues

Issue	Cause	Fix
Grafana shows no data	Wrong Prometheus datasource configured	Reconfigure Datasource URL
No Slack Alerts	Incorrect Webhook	Update Kubernetes secret
API returning 502	Pod failing to load models	Check if the persistent volume is correctly attached to Pod
No metrics in Pushgateway	Service not reachable	Port forward and validate connectivity

Table 1: Common Issues

8 Appendix

8.1 Directory Structure

```
1 devops-assistant-frontend/  
2 ensemble-fastapi/  
3 kubernetes-manifests/  
4 model-training/  
5 prometheus/
```

8.2 Sample .env file

```
1 SLACK_WEBHOOK_URL=<hidden>  
2 MODEL_PATH=/models  
3 PUSHGATEWAY_URL=http://pushgateway.aiops.svc:9091
```

References

- [1] Prometheus Authors (2025). *Prometheus Documentation*. Available at: <https://prometheus.io/docs/>.
- [2] Grafana Labs (2025). *Grafana Documentation*. Available at: <https://grafana.com/docs/>.
- [3] The Kubernetes Authors (2025). *Kubernetes Official Documentation*. Available at: <https://kubernetes.io/docs/>.
- [4] Google Cloud (2025). *Google Kubernetes Engine Documentation*. Available at: <https://cloud.google.com/kubernetes-engine/docs>.
- [5] FastAPI Authors (2025). *FastAPI Documentation*. Available at: <https://fastapi.tiangolo.com/>.
- [6] Docker Inc. (2025). *Docker Documentation*. Available at: <https://docs.docker.com/>.
- [7] Docker Hub (2025). *Docker Hub Repository Service*. Available at: <https://hub.docker.com/>.
- [8] Slack Technologies (2025). *Slack API Documentation*. Available at: <https://api.slack.com/>.