National College of Ireland

Configuration Manual

MSc Research Project
Masters in Cloud Computing

Deepa Shatoji
Student ID: x23427264

School of Computing
National College of Ireland

Supervisor: Abubakr Siddig

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Deepa Shatoji
Student ID:	x23427264
Programme:	Masters in Cloud Computing
Year:	2025
Module:	MSc Research Project
Supervisor:	Abubakr Siddig
Submission Due Date:	10/12/2025
Project Title:	Configuration Manual
Word Count:	735
Page Count:	5

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:	Deepa Ashok Shatoji
Date:	28th January 2026

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

Deepa Shatoji
x23427264

1 Introduction

This configuration guideline records the hardware and software specifications employed in the research study entitled *"Understanding the Impact of Optimised Resource Allocation in Cloud Computing to enhance cost and performance."*

The paper uses machine learning algorithms such as Logistic Regression, Random Forest, XGBoost and Gradient Boosting to evaluate the performance of resource allocation in multi-cloud systems. This report contains full specifications of the system structure, development platform, and deployment processes.

2 System Configuration

2.1 Hardware Configuration

The local development and testing were done on the following hardware setup:

Component	**Specification**
Processor	Apple M3 Chip (Apple Silicon, ARM-based architecture)
Installed RAM	16 GB Unified Memory
Storage	494.38 GB SSD (306.47 GB available)
Graphics Card	Integrated Apple M3 GPU
System Type	Apple Silicon (ARM 64-bit architecture)
Operating System	macOS Sequoia 15.4.1

Table 1: Hardware Configuration of MacBook Air (M3, 2024)

2.2 Cloud Computing Environment

The main development and model training were performed using Google Colaboratory, that offers cloud-based computing infrastructure such as GPUs. TThis platform removes the need to install infrastructure at the local level and offers access to built in machine learning libraries in Python.

3 Software Configuration

The following software libraries and frameworks were used for data processing, machine learning, and visualisation:

Package	Version	Purpose
Python	3.10+	Primary programming language
Pandas	2.2.2	Data manipulation and analysis
NumPy	2.0.2	Numerical computations
Scikit-learn	1.6.1	Machine learning algorithms
XGBoost	3.1.2	Gradient boosting implementation
TensorFlow	2.19.0	Deep learning framework
Keras	3.10.0	Neural network API
Matplotlib	3.10.0	Data visualisation
Seaborn	0.13.2	Statistical visualisation

Table 2: Software Configuration

Installation Command:

```
!pip install pandas numpy scikit-learn xgboost matplotlib seaborn tensorflow keras
!pip install imbalanced-learn shap
```

4 Project Implementation

4.1 Data Configuration

Required Data:-

`multi_cloud_dataset.csv`

Required Field inlcude:

- Service ID, Service Type, Cloud Provider, Edge Node ID
- CPU Utilisation,Memory Usage, Storage Usage
- Network Bandwidth, Service Latency , Response Time
- Throughput, Load Balancing, QoS Score, Workload Variability

The dataset contains 1,000 services instances from four major cloud providers: AWS, Azure, Google Cloud, and IBM.

4.2 Uploading the Dataset in Google Colab

Google colab does not save files between runs and therefore, the data should be uploaded each time it is being run. The next code fragment is the one which makes the user input the file.

`multi cloud dataset.csv` from their local machine:

```
from google.colab import files

print("Please select multi_cloud_service_dataset.csv from your computer...")
uploaded = files.upload()   # choose the CSV file when the dialog appears
```

Figure 1: Upload Dataset

* Colab then confirms the upload like in the following:

```
Please select multi_cloud_service_dataset.csv from your computer...
Browse... multi_cloud_service_dataset.csv
multi_cloud_service_dataset.csv(text/csv) - 166456 bytes, last modified: n/a - 100% done
Saving multi_cloud_service_dataset.csv to multi_cloud_service_dataset.csv
```

Figure 2: Dataset Uploaded

* After uploading, the file is familiar in the colab, working directory can be loaded with:

```
# Load the dataset
import pandas  as pd
df = pd.read_csv('multi_cloud_service_dataset.csv')
```

5 Execution Workflow

Run the following programs in this order:

1. Install Dependencies
2. Load Dataset
3. Exploratory Data Analysis (EDA)
   – Correlation Heatmaps
   – Feature distributions
4. Feature Engineering and KPI Generation
5. Preprocessing (Label Encoding, StandardScaler)
6. Hyperparameter Tuning
7. Model Training
   – Logistic Regression
   – Random Forest
   – XGBoost
   – Gradient Boosting
8. Model Evaluation
   – Accuracy, Precision, Recall, F-score
   – Confusion Matrix

9. KPI Calculation
10. Cost- Performance Analysis
11. CloudResourceAllocator Demonstration

The notebook automatically generates all metrics and plots.

5.1 Reproducibility Notes

- Use Python 3.10 in Colab.
- Keep original dataset column names.
- Run notebook cells in order.
- Use the same scaler and encoder pipelines defined in CODE.ipynb.

5.2 Troubleshooting

Issues	**Cause**	**Solution**
Model accuracy changes	Random seed not fixed	Use `random_state=42`
Import error	Missing dependency	Run installation cell again
Runtime crash	Colab memory limit	Restart runtime

5.3 Results Summary

Model	**Accuracy**	**Precision**	**F1-Score**
Logistic Regression	96.0%	96.12%	95.96%
Gradient Boosting	90.5%	90.37%	90.40%
XGBoost	90.0%	89.86%	89.82%
Random Forest	86.0%	86.45%	85.96%

Table 3: Results Summary

The test indicates that the Logistic Regression model was the most effective as it was able to do so. performed with the best accuracy (96.0%), which shows that simple models can work. surpassedly well with well-designed features in the allocation of cloud resources. tasks. The accuracy and F1- score of Gradient Boosting stood at 90.5 and 90.40 respectively. and XGBoost had 90.0% and 89.82 respectively. Random Forest was found to have a. relatively worse quality of results since it had an 85.96 F1-score with an accuracy of 86.0%. Generally, this is similarity in accuracy and precision, recall and F1-score demonstrates that the. There was consistency in the manner in which classifiers differentiated amongst categories of efficiency across the. models. The superior functioning of the Logistic Regression is also associated with superior linear. separability of features and labels of this dataset that are engineered.

5.4 Conclusion

This configuration guide will make sure that examiners, researchers and cloud engineers will be able to. exactly recreate the machine learning pipeline and CloudResourceAllocator system using the exact configuration of the system. as implemented. The manual meets the rubric requirements of reproducibility, structure. preciseness, and technical exactness.

6 Repositories and Dataset Links

- Kaggle: `https://www.kaggle.com/`
- Dataset: `https://www.kaggle.com/datasets/ziya07/multi-cloud-service-composition-da`

7 References

- Python: `https://www.python.org`

References