National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Parth Moradiya
Student ID: x21199434

School of Computing
National College of Ireland

Supervisor: Rejwanul Haque

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Parth Ishvarbhai Moradiya
Student ID:	21199434
Programme:	MSc in Cloud Computing
Year:	2024-25
Module:	MSc Research Project
Supervisor:	Rejwanul Haque
Submission Due Date:	29th January, 2025
Project Title:	Configuration Manual
Word Count:	1105
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:	Parth Ishvarbhai Moradiya
Date:	29th January, 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

Parth Moradiya
Student ID: x21199434

1 System Requirements

1.1 Hardware Requirements

- Processor: Intel Core 10th Gen Core-i5 or above
- RAM: 16 GB or above
- Storage: SSD/HDD with minimum 50 GB free space
- Network: Active Internet Connection for code packages and dependency downloads

1.2 Software Requirements

- OS: Windows 10/11/ Ubuntu Linux / Mac
- Java Development Kit (JDK) 11
- IntelliJ Idea Community Edition
- Python 3.8 or higher
- Git (for version control) - Optional

2 Development Environment Setup

2.1 IntelliJ IDEA Installation

1. Download IntelliJ IDEA Community Edition from JetBrains website: https://www.jetbrains.com/idea/download/
2. Run the installer and complete the installation
3. During first launch, select to install the following plugins:
    - Python Community Edition
    - Maven Integration

2.2 JDK Configuration in IntelliJ

1. Open IntelliJ IDEA
2. Navigate to File → Project Structure → Platform Settings → SDKs
3. Click + and select "Download JDK"
4. Choose version 11 and vendor (Oracle OpenJDK)
5. Click Download and wait for it to complete successfully

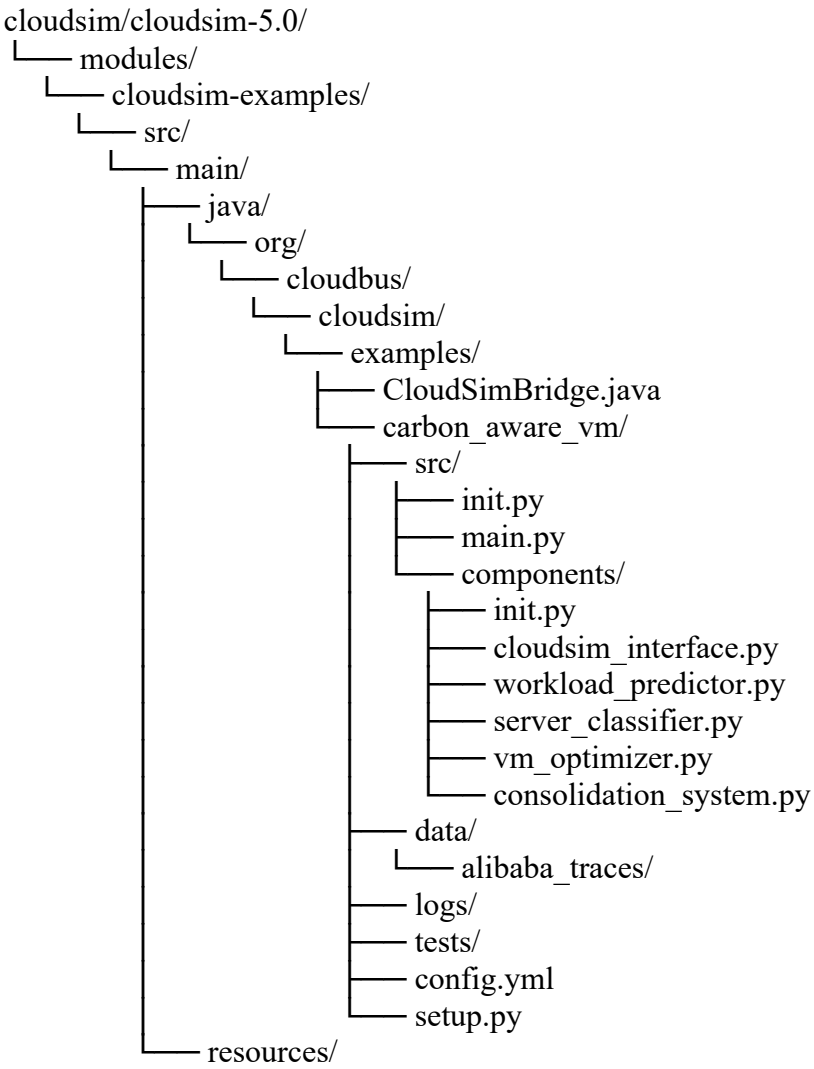
2.3 Python Configuration in IntelliJ

1. Navigate to File → Project Structure → Platform Settings → SDKs
2. Click + and select "Add Python SDK"
3. Choose "New environment using Virtualenv"
4. Select Python 3.8 or higher as base interpreter

5. Click OK to create the environment

3 Development Environment Setup

3.1 Project Structure



3.2 Project Import

1. File → Open
2. Navigate to cloudsim/cloudsim-5.0 directory
3. Select and open as project
4. Wait for IntelliJ to index and download dependencies

3.3 Maven Configuration

1. Load the project folder
2. In the Maven tool window (View → Tool Windows → Maven):
   - Expand project
   - Right-click on project root
   - Select "Download Sources"
3. Verify Maven configuration:
   - Open root pom.xml

- Click the Maven refresh button in the top-right
- Wait for dependencies to download

3.4 Install Python Dependencies

- Open Terminal tool window in IntelliJ (View → Tool Windows → Terminal)
- Ensure your Python virtual environment is activated
- Install dependencies via IntelliJ's integrated terminal:

```
cd modules/cloudsim-examples/src/main/java/org/cloudbus/cloudsim/examples/carbon_aware_vm
pip install -e .
```

4 Project Configuration

4.1 CloudSim Bridge Setup

- Verify `resources` directory contains required JARs:
  1. cloudsim4.0.jar
  2. py4j.jar

- Configure CloudSim Bridge run configuration:

  1. Main class: `org.cloudbus.cloudsim.examples.CloudSimBridge`
  2. Working directory: `modules/cloudsim-examples`
  3. Module: `cloudsim-examples`

4.2 Python Component Setup

- Configure Python run configuration:

  - Script path: `carbon_aware_vm/src/main.py`
  - Working directory: `carbon_aware_vm`
  - Python interpreter: Previously created virtualenv

4.3 Configure Data Directory

- Place Alibaba trace dataset in: `carbon_aware_vm/data/alibaba_traces/`
- Create logs directory: `carbon_aware_vm/logs/`

5 Project Execution

- Start CloudSim Bridge:

  - Select CloudSimBridge configuration
  - Run (Shift+F10)

- Run Carbon-Aware VM System:

  - Select Python configuration for main.py
  - Run (Shift+F10)

6 Project Components

- CloudSim Bridge modifications in: examples/CloudSimBridge.java
- Core components in carbon_aware_vm/src/components/
- data/: Dataset and input files
- setup.py: Python package configuration
- CloudSimBridge.java: Java-Python interface
- cloudsim_interface.py: CloudSim communication
- workload_predictor.py: LSTM implementation
- server_classifier.py: Load classification
- vm_optimizer.py: NSGA-II optimization
- consolidation_system.py: Main system logic
- main.py: function calls to call other components and result evaluation

7 Sample Code Simulation

```
def main():

    # Initialize the system
    system = VMConsolidationSystem()
    system.initialize()

    # Load data from the cluster_traces.csv file
    data =
pd.read_csv('carbon_aware_vm/data/alibaba_traces/cluster_trace
s.csv')
  print(data.head())

    # Run optimization
    system.run_optimization(data)

    # Monitor metrics
    system.monitor_metrics()

if __name__ == "__main__":
  main()
```

8 Execution Results

8.1 Simulation Initialization

- components.cloudsim_interface - INFO - Successfully connected to Carbon-Aware CloudSim
- components.cloudsim_interface - INFO - Initialized CloudSim with 10 VMs
- components.consolidation_system - INFO - VM Consolidation System initialized

8.2 Training Progress Monitoring

Epoch 1/100

56/56 ━━━━━━━━━━━━━━━━━━━━ 3s 20ms/step - loss: 0.3560 - val_loss: 0.1966

...

Epoch 41/100

56/56 ━━━━━━━━━━━━━━━━━━━━ 1s 14ms/step - loss: 0.0866 - val_loss: 0.0980

The LSTM training progress should show:

- Approximately 41 epochs of training
- Loss values decreasing over time
- Training time of ~1s per epoch
- Final validation loss around 0.098

8.3 VM Optimization Process

The VM optimization process should show 75 generations completing,

- components.vm_optimizer - INFO - Generation 1/75 completed

...

- components.vm_optimizer - INFO - Generation 75/75 completed

8.4 Migration Operations

Successful Migrations will be logged as,

- components.cloudsim_interface - INFO - Attempting to migrate VM 0 from host 0 to 3
- components.cloudsim_interface - INFO - Successfully migrated VM 0
- components.consolidation_system - INFO - Successfully migrated VM 0 from host 0 to 3

8.5 Final Performance Metrics

Current system metrics:

- Energy consumption: ~5.89 kWh
- CO2 emissions: ~2.11 kg
- Number of migrations: ~9
- Prediction accuracy: ~0.85
- Classification accuracy: ~1.00