

Configuration Manual

MSc Research Project
MSc in Artificial Intelligence

Shraddha Shailesh Gawade
Student ID: x23408324

School of Computing
National College of Ireland

Supervisor: Dr. Anu Sahini

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: Shraddha Shailesh Gawade
Student ID: 23408324
Programme: MSc in Artificial Intelligence **Year:** 2025
Module: Research in Computing
Lecturer: Dr. Anu Sahini
Submission Due Date: 11-12-2025
Project Title: Multi-Industry Staff Scheduling Under Labor Law Constraints Using MILP, CP and Genetic Algorithms

Word Count: 675

Page Count: 7

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: Shraddha Shailesh Gawade

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

Shraddha Shailesh Gawade
Student ID: x23402324

1 System Configuration

1.1 Hardware Specification

The project was implemented and evaluated on a personal machine with the following configuration:

- Device Name: Shraddha
- Processor: 13th Gen Intel Core i5-13335U, 1.30 GHz
- RAM: 16 GB
- System Type: 64-bit OS, x64-based processor

This hardware specification was useful for running optimization models such as MILP, CP and GA for staff scheduling.

1.2 Operating System

- OS Version: Windows 11
- Version: 24H2

1.3 Software Configuration/ Environment

- Programming Language: Python 3.10.11
- IDE / Notebook: Jupyter Notebook
- Package: pip

1.4 Libraries Used

The libraries were used for this project are given below with their purpose:

- Pandas and NumPy - Data Preprocessing
- PuLP (CBC solver) – MILP Optimization
- ortools.sat.python.cp_model – Constraint Programming Optimization
- DEAP – Genetic Algorithm Optimization
- Matplotlib – Visualization
- DateTime – Date/Time Processing
- Os, Random – OS utilities

Libraries should have to install using:

`pip install pandas numpy pulp ortools deap matplotlib`

```

import os
import random
from datetime import datetime, timedelta

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from pulp import (
    LpProblem, LpVariable, LpBinary, LpSum,
    LpMaximize, PULP_CBC_CMD
)
from ortools.sat.python import cp_model
from deap import base, creator, tools, algorithms

```

2 Dataset Acquisition and Preparation

2.1 Data Sources

1. Restaurant Dataset

- Real dataset collected ethically from my workplace (Anjappar Chettinad Restaurant).
- This dataset includes staff names, roles, availability, shift assignments (start and end) and contract type.

2. Healthcare Nurse Scheduling Dataset

- Downloaded from public research repository from Kaggle.
- Link of the dataset:

3. Retail Workforce Dataset

- Download from the public research repository from Kaggle.
- Link for the dataset:

2.2 Dataset Merging

All these three datasets were merged using a python module into a one single dataset that:

- Normalizing columns across these three different industries.
- Standardized contract hours, shift and preference score.
- Merged dataset to get new dataset named as combined_dataset.csv stored at data/processed/combined_dataset.csv.
- This dataset contains totally 6372 rows including all three industries.

2.3 Exploratory Data Analysis

EDA was done withing Jupyter Notebook using below:

- Frequency distribution of industries, shift patterns and roles.
- Pandas.describe() for statcal summaries of dataset.
- Visualization to check staff demand and contract types.

3 System Workflow

3.1 First load the dataset

Keep the paths ready for two main csv files such as combined dataset and next week staff availability.

```
# Project paths
COMBINED_FILE = "../data/processed/combined_dataset.csv"
NEXT_WEEK_FILE = "../data/processed/next_week_availability.csv"

os.makedirs("../data/processed", exist_ok=True)

print("Combined dataset path:", COMBINED_FILE)
print("Next week availability path:", NEXT_WEEK_FILE)

combined = pd.read_csv(COMBINED_FILE)
print("Rows in combined dataset:", len(combined))
combined.head()
```

3.2 Next Week Availability Generation

- A realistic availability for one week (7 days) generated using python:
- Part time employees assigned with 3 working days (20 hours).
- Full time employees assigned with 5 working days (40 hours).
- Based on Industry type availability applied.

Output File contains 7 days of staff availability for each industry includes total 7455 rows.
Output file saved as next_week_availability.csv in a
data/processed/next_week_availability.csv.

3.3 Decide actual scheduled start/end time for an assigned shift

Industry-specific availability windows**:

- Restaurant:
 - FT: 11:30–19:30
 - PT: either 11:30–15:30 or 17:00–21:00
- Healthcare:
 - Rotating: 07:00–15:00, 15:00–23:00, 23:00–07:00
- Retail:
 - One of: 09:00–13:00, 13:00–17:00, 13:00–21:00

```
def pick_shift(industry, contract_type, date_str, dates_index):
    """Decide actual scheduled start/end time for an assigned shift, based on industry, C/P and weekend"""
    ct = str(contract_type).upper()
    dt = pd.to_datetime(date_str)
    is_weekend = dt.weekday() >= 5

    if industry == "Restaurant":
        if ct == "FT":
            return "11:30", "19:30"
        else:
            if is_weekend:
                return "18:00", "22:00"
            else:
                return "11:30", "15:30"

    elif industry == "Healthcare":
        idx = dates_index[date_str] % 3
        if idx == 0:
            return "07:00", "15:00"
        elif idx == 1:
            return "15:00", "23:00"
        else:
            return "23:00", "07:00"

    else: # Retail
        if ct == "FT":
            if is_weekend:
                return "13:00", "21:00"
            else:
                return "09:00", "18:00"
        else:
            return "13:00", "17:00"
```

3.4 Execution of Scheduling Models

Three scheduling models were train and tested or executed given as below:

- Mixed Integer Linear Programming (MILP) – using CBC solver
- Constraint Programming (CP-SAT) – using Google OR-Tools
- Genetic Algorithm (GA) – Using DEAP framework of evolution

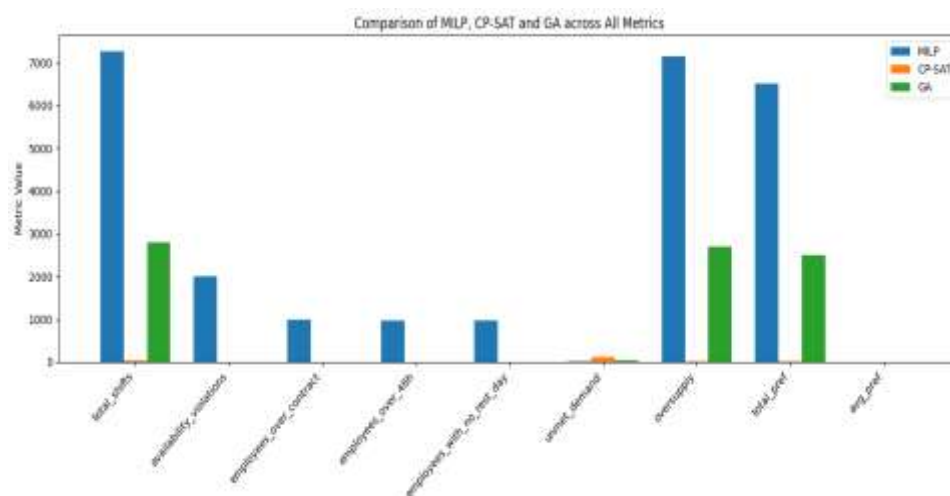
Each model produces a new week staff roster with the help of historical staff roster with staff start and end time, role-specific, labour compliance, constraints, availability.

Output was stored in the format of csv file such as:

/data/processed/schedules_milp.csv

/data/processed/schedules_cp.csv

/data/processed/schedules_ga.csv



3.5 Evaluation

Evaluation of model was done based on below metrics:

- Total assigned shifts, Employees over contract, Employees with more than 48 hours, Unmet demand, Oversupply, Availability violations, Preference Score

	total_shifts	availability_violations	employees_over_contract	employees_over_48h
MILP	7276.0	1996.0	983.0	972.0
CP-SAT	31.0	7.0	0.0	0.0
GA	2799.0	0.0	0.0	0.0

employees_with_no_rest_day	total_pref	avg_pref	unmet_demand	oversupply
972.0	6514.1	0.895286	12.0	7148.0
0.0	27.9	0.900000	121.0	12.0
0.0	2505.8	0.895248	36.0	2695.0

Step To Follow:

1. Install python 3.10.11
2. Install required libraries given above.
3. Place the raw datasets in /data/raw/ folder.
4. Run merge.py model to get combined dataset.
5. Generate Next week availability.
6. Execute each scheduling model.
7. Visualize and evaluate the results.

Restaurant datasets were handled ethically under project guidelines.

References

PuLP Library Mitchell, S., 2011. PuLP: A Python Linear Programming Toolkit. Available at: <https://coin-or.github.io/pulp/>

Google OR-Tools Perron, L. & Furnon, V., 2023. OR-Tools: Google Optimization Tools. Available at: <https://developers.google.com/optimization>

DEAP Evolutionary Algorithm Framework Fortin, F.-A. et al., 2012. DEAP: Distributed Evolutionary Algorithms in Python. Available at: <https://deap.readthedocs.io/>

Healthcare Staff Scheduling Dataset (Kaggle) Wasis, K., 2023. Healthcare Staff Scheduling Dataset. Kaggle. Available at: <https://www.kaggle.com/datasets/kenwasisa/healthcare-staff-scheduling>

Retail Employee Schedule Dataset (Kaggle) Boss, 2022. Employee Schedule Dataset. Kaggle. Available at: <https://www.kaggle.com/datasets/bossssss/employee-schedule>