Community Service Request and Response Management System with ML-Based Priority Ranking

AHybridMachineLearning and Knowledge-Based Reasoning Project

Collaborative Final Project
CSST101 – Machine Learning
CSST102 – Knowledge Representation and Reasoning

Submitted by: Wilmark Veridiano
Group Name: Barangay

Group Members: -
Eleazar, Cj
Tobias, Inigo
Coronacion, John Christian
Veridiano, Wilmark

Instructor: Mr. Mark P. Bernardino

Date Submitted: _____Jan 2 2026_____

PROJECT OVERVIEW

Our system manages community service requests, automatically predicting the priority of requests using Machine Learning, and enhancing decisions with rule-based reasoning to ensure accurate, realistic prioritization.
It helps local authorities respond efficiently and safely to service requests.

OBJECTIVES

General Objective: Develop a hybrid AI system to classify community service requests by priority and provide actionable recommendations.

Specific Objectives: -
Build a Machine Learning model to predict request priority (High, Medium, Low).

Design Knowledge-Based Rules to adjust predictions based on real-world logic.

Implement a user-friendly web interface and API for submitting requests.

SYSTEM ARCHITECTURE

User Input → Machine Learning Model → KRR Rules → Final Priority → Recommendations

1. **Users submit service requests through a web form .**
2. **The ML model predicts an initial priority based on data.**
3. **KRR rules adjust priority if required**
4. **The system outputs the final priority and recommended actions.**

MACHINE LEARNING COMPONENT (CSST101)

Algorithm Used: Random Forest Classifier
Dataset Size: 500 synthetic community service requests
Model Accuracy: 90%

MACHINE LEARNING PIPELINE

Data Collection:
Synthetic dataset of community service requests from Brgy. Bagumbayan.

Data Preprocessing: _Handled missing values, calculated, encoded categorical variables.

Model Training:
Handled missing values, calculated days_open, encoded categorical variables.

Model Evaluation:

- Split dataset into 80% train and 20% test.
- Accuracy ~90%, classification report validated priority predictions.

Model Deployment:
Exported model with pkl for Flask web application.

DATASET DESCRIPTION

Dataset Type: Synthetic community service requests CSV
Number of Records: 500
Target Variable: High, Moderate, Low

KNOWLEDGE REPRESENTATION & REASONING (CSST102)

Rule 1: IF urgency_level = Urgent THEN priority = High
Rule 2: IF severity_level = Severe THEN priority = High

Rule 3: IF impact_scope > 100 THEN increase priority
(Low → Moderate, Moderate → High)

Rule 4: IF location_type = Highway AND time_reported = Night THEN priority = High

Rule 5: IF weather_condition = Storm AND request_type = Road Damage THEN priority = High

HYBRIDDECISION LOGIC

1. ML model predicts initial priority.
2. Rules check for extreme cases or inconsistencies.
3. Final priority is determined by blending ML prediction and rule-based adjustments.
4. Recommendations generated based on final priority.

SYSTEMFEATURES

Wellness risk prediction ✓
Rule-based recommendations ✓
Web interface / API ✓
Google Colab deployment ✓

TESTINGAND EVALUATION

Test Case | Input Summary | Expected Output

1	road dmg, urgent, severe, 50, highway, morning, normal, 1	High, High
2	water leakage, low, minor, 5, residential area, morning, normal, 3.	Moderate, Moderate
3	garbage collection, low, minor, 10, school zone, morning, normal, 1,	Low, Low

CONCLUSION

The Community Service Request and Response Management System successfully integrates Machine Learning and Knowledge-Based Reasoning, producing accurate priority predictions and actionable recommendations. This hybrid approach ensures safe, reliable, and explainable decision-making for community service management.

GROUP CONTRIBUTION

Member Name | Contribution
Veridiano, Wilmark
ML model development & training
Eleazar, Cj
Flask web interface and API integration
Tobias, Inigo
Rule-based reasoning design
Coranacion, John Christian
Rule-based reasoning design

REFERENCES

- scikit-learn documentation: https://scikit-learn.org
- Python Pandas documentation: https://pandas.pydata.org
- Flask documentation: https://flask.palletsprojects.com
- Hybrid AI design principles