

User Manual - Machine Learning Platform for Intelligent Water Systems Management - Final Project

Robert Castro
Calvin Chau
Yvan Michel Kemsseu Yobeu
Laila Velasquez
Kassandra Vera

Friday, May 9, 2025

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Overview	3
2	Jira Link	3
3	Formal Objective Breakdown	3
4	Goals and Importance	3
5	Installation and Setup	3
5.1	System Requirements	3
5.2	Installation Steps	3
5.3	Docker Configuration	4
6	User Interface Guide	4
6.1	Dashboard Overview	4
6.2	Analytics Page	4
6.3	Settings	4
6.4	Reports	4
7	Usage Instructions	4
7.1	Logging In	4
7.2	Viewing Data	4
7.3	Configuring Alerts	4
7.4	Generating Reports	4
8	Troubleshooting	5
8.1	Common Issues	5
8.2	Error Messages	5
9	Appendix	5
9.1	API Documentation	5
9.2	Database Schema	5

1 Introduction

1.1 Purpose

This User Manual provides step-by-step guidance for users to effectively navigate and utilize the Intelligent Water Systems Management Platform.

1.2 Overview

The platform allows users to monitor water consumption, detect anomalies, and optimize water usage through real-time data visualization and predictive analytics.

2 Jira Link

You can access the Jira board for project management and sprint planning here: <https://csula-kv.atlassian.net/jira/software/projects/WSM/boards/6>

3 Formal Objective Breakdown

The primary objectives of this platform are:

- Real-time monitoring of water consumption
- Automated alerts for leak detection
- Predictive analysis for maintenance
- Detailed analytics for optimizing water usage

4 Goals and Importance

This platform is designed to bridge the gap in water consumption monitoring by providing:

- Enhanced water usage efficiency
- Early detection of leaks to prevent waste
- Data-driven insights for better resource management

By offering real-time monitoring and predictive analytics, it empowers municipalities and property owners to optimize water consumption and reduce waste.

5 Installation and Setup

5.1 System Requirements

- Docker (latest version)
- Node.js (for frontend)
- SQL Database

5.2 Installation Steps

1. Clone the repository from GitHub.
2. Navigate to the project directory.
3. Run `docker-compose up` to initialize all services.
4. Open the application at `http://localhost:5000`.

5.3 Docker Configuration

The Docker setup includes:

- **Frontend:** React-based dashboard
- **Backend:** Flask for API management
- **Database:** SQL instance for data storage

6 User Interface Guide

6.1 Dashboard Overview

The main dashboard displays:

- Real-time water consumption data
- Alerts for abnormal usage
- System health status

6.2 Analytics Page

- Graphical representation of historical data
- Comparative analysis of water usage trends

6.3 Settings

- Sensor configurations
- Alert preferences

6.4 Reports

- Generate and export reports in CSV or PDF formats

7 Usage Instructions

7.1 Logging In

1. Open the application at `http://localhost:5000`.
2. Enter your username and password.
3. Click **Login**.

7.2 Viewing Data

Navigate to the **Dashboard** to see live metrics.

7.3 Configuring Alerts

Go to **Settings** and enable notifications for high usage or leaks.

7.4 Generating Reports

Click on **Reports** and select the time range for data.

8 Troubleshooting

8.1 Common Issues

- **Docker container not starting** → Verify Docker installation and run `docker-compose up --build`.
- **Dashboard not loading** → Clear browser cache and retry.

8.2 Error Messages

- **500 Internal Server Error** → Restart Docker containers.
- **404 Not Found** → Check Docker logs for missing dependencies.

9 Appendix

9.1 API Documentation

Refer to the API endpoints for detailed usage.

9.2 Database Schema

SQL-based with normalized tables for optimized querying.