City Resource and Infrastructure Management System (CRIMS)

1. Preface

This document specifies the software requirements for the City Resource and Infrastructure Management System (CRIMS).
It is intended for system users, developers, testers, and evaluators to understand what the system should do.

2. Introduction

2.1 Purpose

The purpose of this document is to define the functional and non-functional requirements of CRIMS, based on stakeholder needs and system modeling using use-case and activity diagrams.

2.2 Scope

CRIMS supports city authorities in:

- Monitoring infrastructure
- Managing work orders
- Allocating budgets
- Scheduling and executing maintenance
- Tracking performance and reporting

Citizen complaint handling is treated as a separate subsystem under the same domain.

2.3 Definitions

Term	Description
CRIMS	City Resource and Infrastructure Management System
Work Order	Authorized request to perform maintenance
Stakeholder	User interacting with the system

---

3. Overall Description

3.1 System Users

- City Administrator
- Department Officer
- Field Engineer

3.2 Product Perspective

CRIMS is a centralized management system that integrates:

- Monitoring
- Approval
- Scheduling
- Execution
- Reporting

It is not responsible for low-level hardware control.

4. User Requirements (High-Level)

UR1

The system shall allow the City Administrator to evaluate city project performance.

UR2

The system shall allow the City Administrator to approve work orders and allocate budgets.

UR3

The system shall allow the Department Officer to schedule inspections and track task progress.

UR4

The system shall allow the Field Engineer to inspect infrastructure and execute work orders.

---

5. Functional Requirements (System Requirements)

5.1 City Administrator Functions

- FR1: The system shall allow the administrator to monitor city resource usage.
- FR2: The system shall allow the administrator to approve work orders.
- FR3: The system shall allow the administrator to allocate budgets.
- FR4: The system shall generate performance reports.

5.2 Department Officer Functions

- FR5: The system shall allow the officer to review assigned tasks.
- FR6: The system shall allow the officer to report task progress.

- FR7: The system shall allow the officer to schedule periodic maintenance inspections.

5.3 Field Engineer Functions

- FR8: The system shall allow engineers to inspect infrastructure.
- FR9: The system shall allow engineers to request required resources.
- FR10: The system shall allow engineers to identify and report new issues.
- FR11: The system shall allow engineers to execute approved work orders.
- FR12: The system shall allow engineers to perform maintenance activities.

---

6. Non-Functional Requirements

6.1 Performance Requirements

- NFR1: The system shall respond to user actions within **3 seconds**.
- NFR2: Report generation shall complete within **5 seconds**.

6.2 Security Requirements

- NFR3: Only authorized users shall access system functions.
- NFR4: Role-based access control shall be enforced.

6.3 Reliability Requirements

- NFR5: The system shall ensure data consistency for work orders.
- NFR6: The system shall recover gracefully from system failures.

6.4 Usability Requirements

- NFR7: The system shall provide a simple and intuitive interface.
- NFR8: Users shall require minimal training to operate the system.

---

7. System Models

This system is modelled using:

- **Use-Case Diagram** – to represent user goals
- **Activity Diagrams** – to represent workflow and decision logic
- **Sequence Diagram**

---

8. System Evolution

This chapter describes the fundamental assumptions and anticipated changes to help system designers avoid restrictive decisions.

8.1 Fundamental Assumptions

- **Administrative Hierarchy**: The system assumes a three-tier administrative structure (Administrator, Officer, Engineer) for all workflow logic.
- **Connectivity**: It is assumed that Field Engineers have intermittent access to data networks while on-site.

8.2 Anticipated Changes

- **Hardware Evolution**: Integration with **IoT sensor networks** for automated infrastructure health monitoring.

- **Evolving User Needs**: Future requirements for **automated predictive maintenance** based on historical performance reports.

---

9. Appendices

The appendices provide detailed, specific information regarding the hardware and data organization.

9.1 Hardware Requirements

Following the requirement for a centralized management system, the hardware must meet these configurations:

- **Server (Minimal)**: 8-core CPU, 32GB RAM, 1TB SSD for the centralized database.
- **Field Devices**: Mobile tablets with GPS capabilities and a minimum of 8 hours of battery life to support on-site inspections.

9.2 Database Requirements

This section defines the logical organization of data used by CRIMS:

- **Entity Relationships**: Each **Work Order** must be uniquely linked to a **Budget Allocation** and an assigned **Field Engineer**.

- **Persistence**: Historical data for all city locations must be preserved to allow for duplicate issue analysis.

10. Index

To ensure scannability for different stakeholders, the following indexes are included:

- **Index of Diagrams**:

- **Index of Functions**:
  - City Administrator Functions (FR1–FR4)
  - Department Officer Functions (FR5–FR7)
  - Field Engineer Functions (FR8–FR12)

11. Assumptions and Constraints

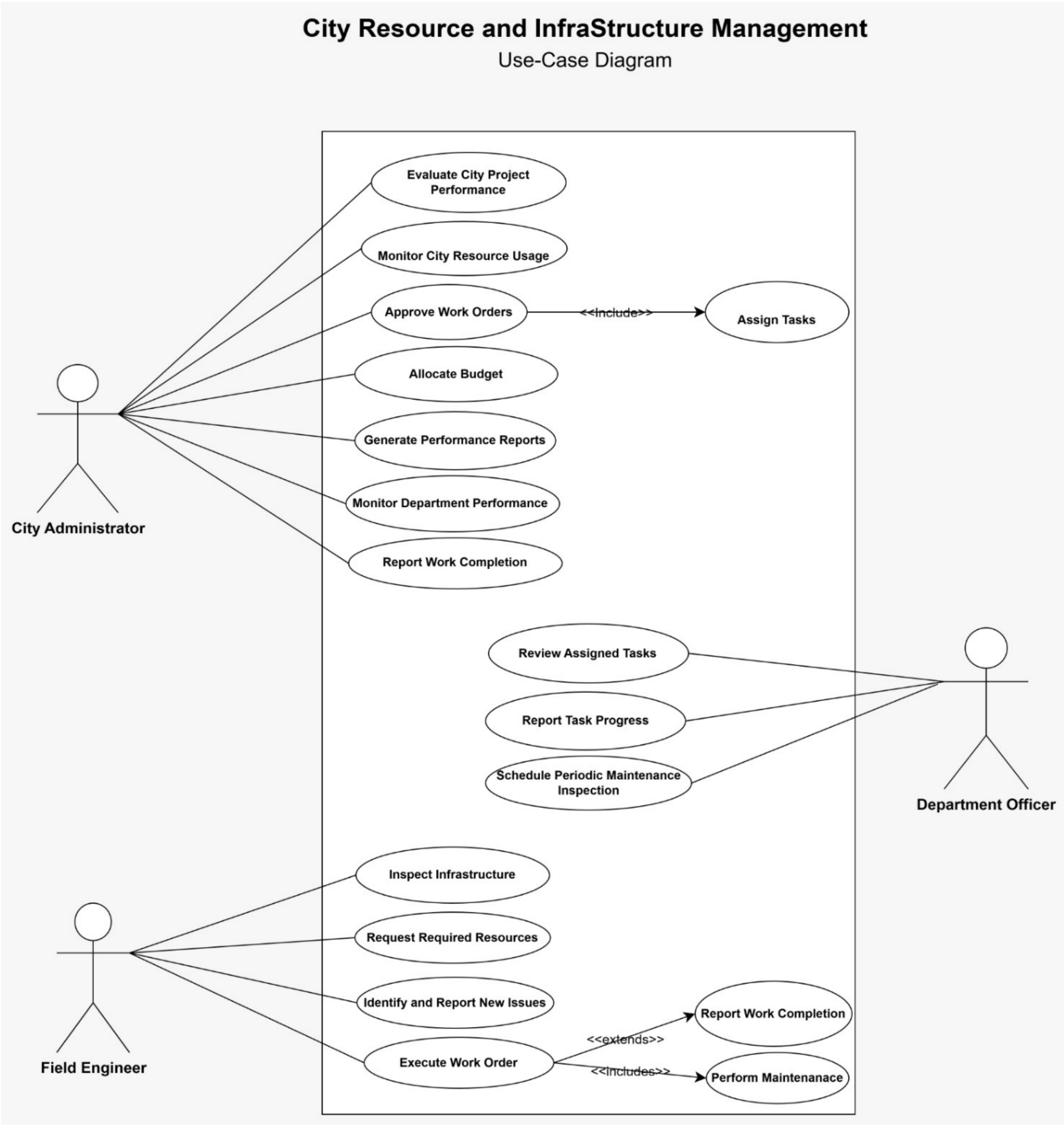
- The system operates within municipal administrative policies.
- Budget approvals follow predefined authority levels.
- Citizen complaint handling is modeled separately.

---

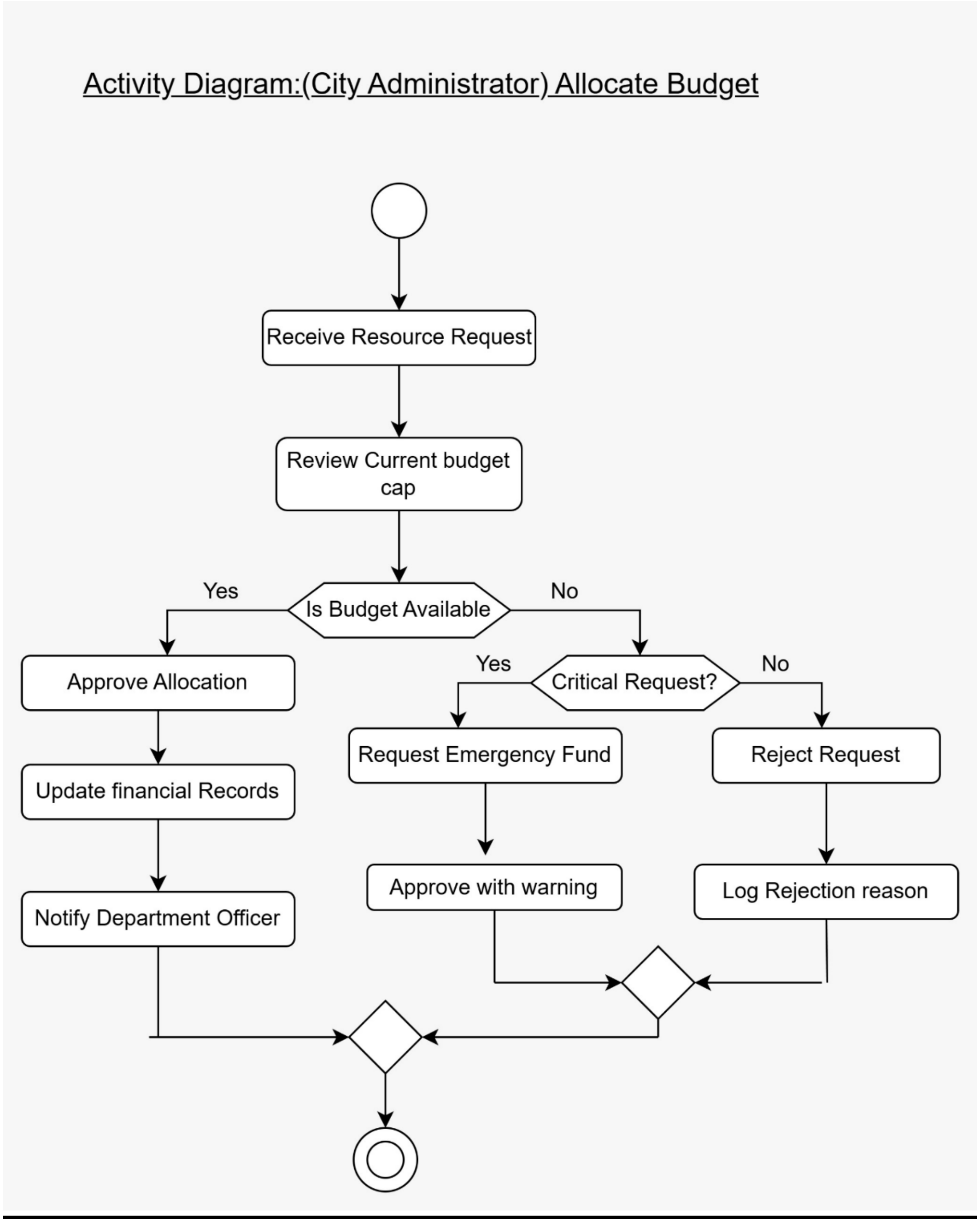
12. Conclusion

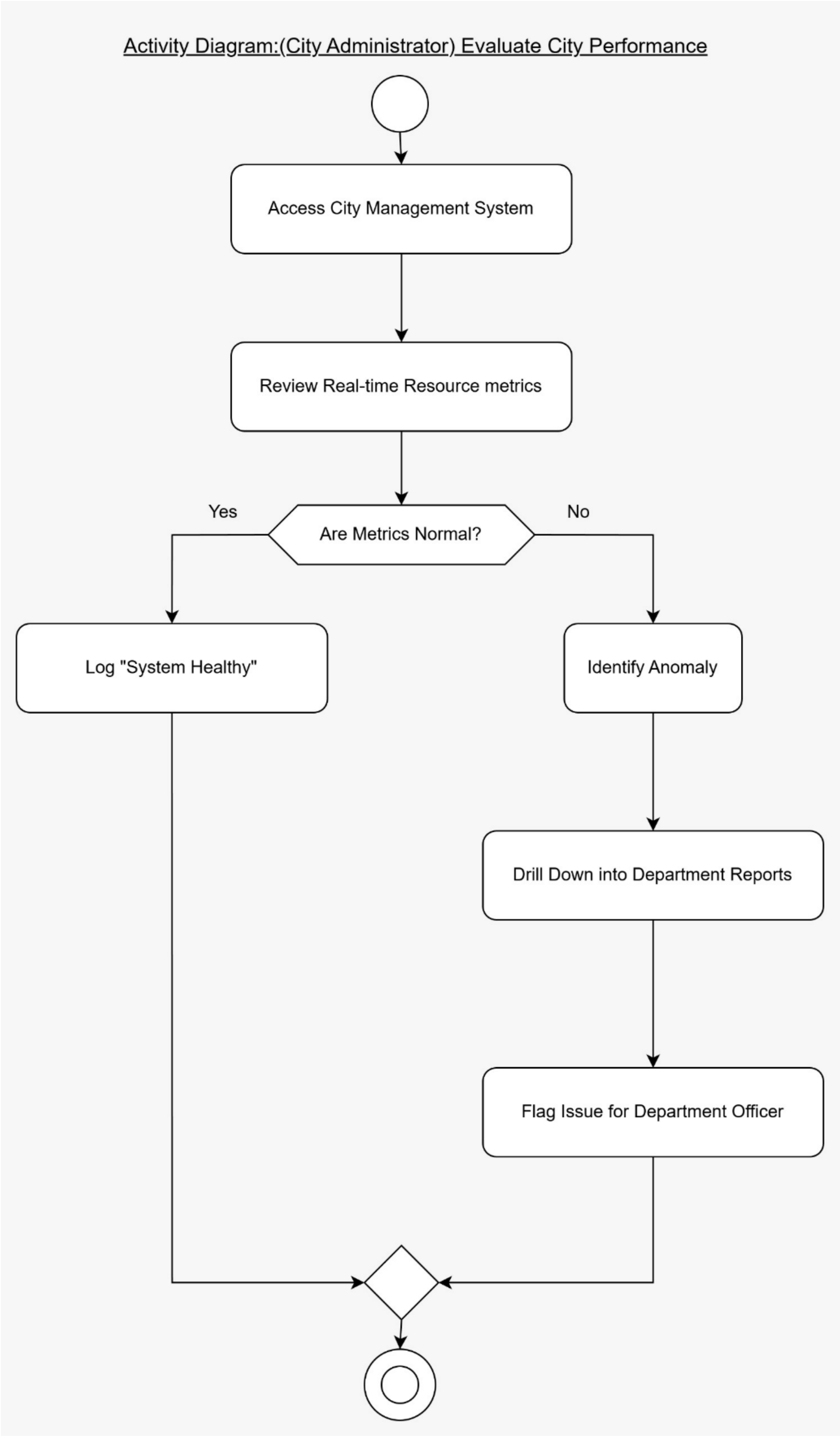
This SRS defines the functional scope and constraints of CRIMS, ensuring clarity, consistency, and traceability with UML models.

Use Case Diagram

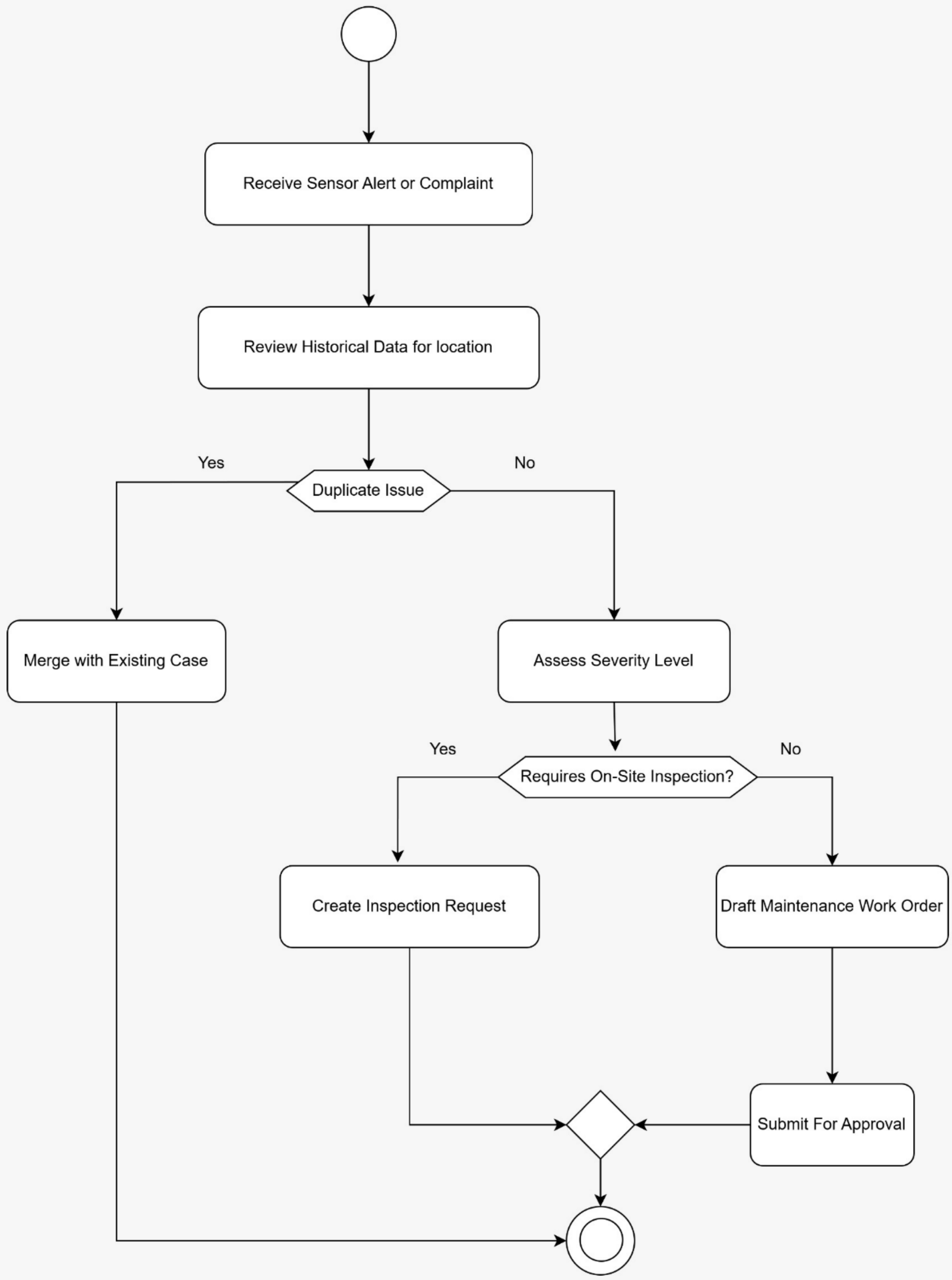


Activity Diagrams

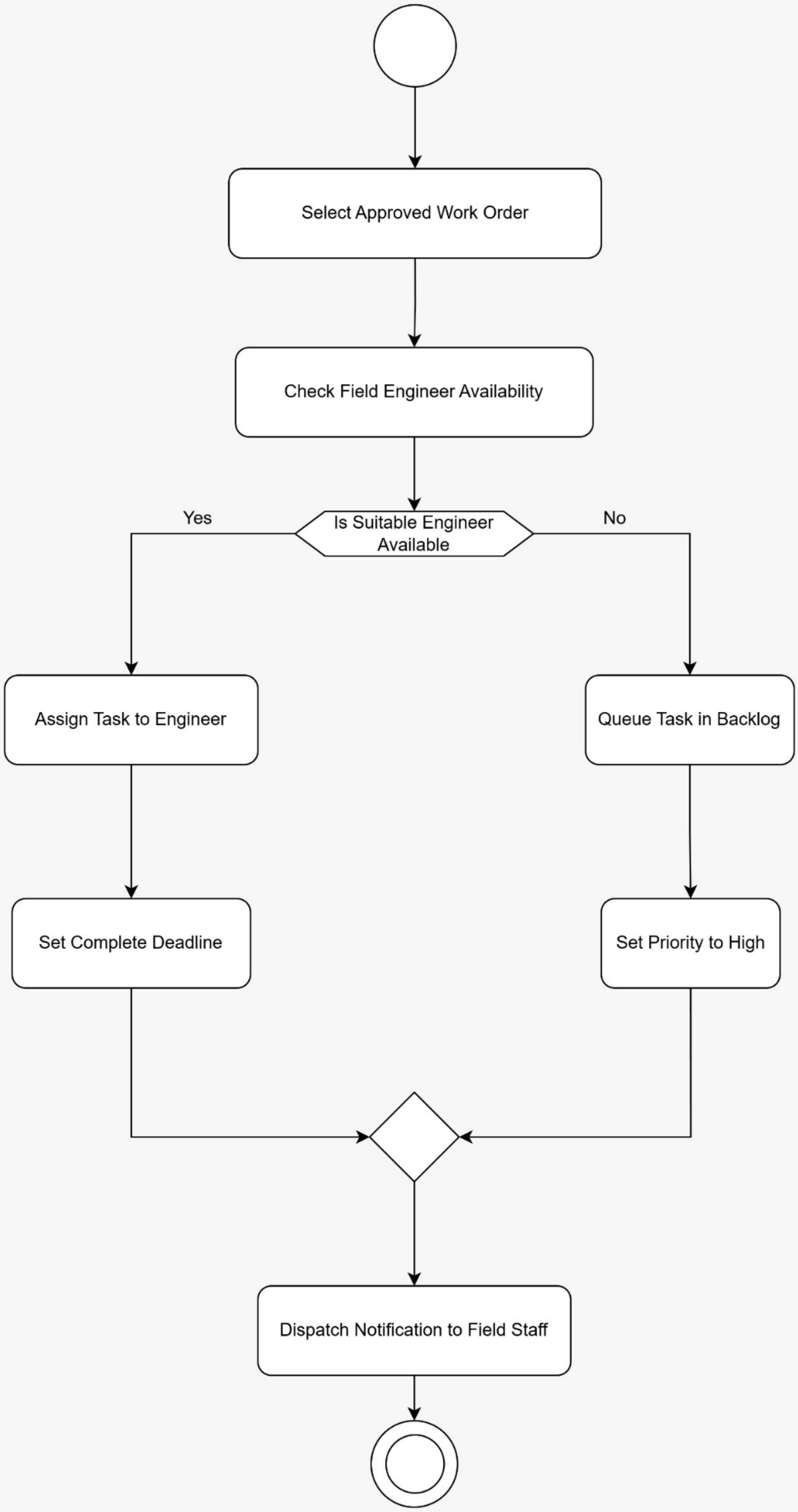



Activity Diagram:(City Administrator) Evaluate City Performance
Access City Management System
Review Real-time Resource metrics
Are Metrics Normal?
Yes
No
Log "System Healthy"
Identify Anomaly
Drill Down into Department Reports
Flag Issue for Department Officer

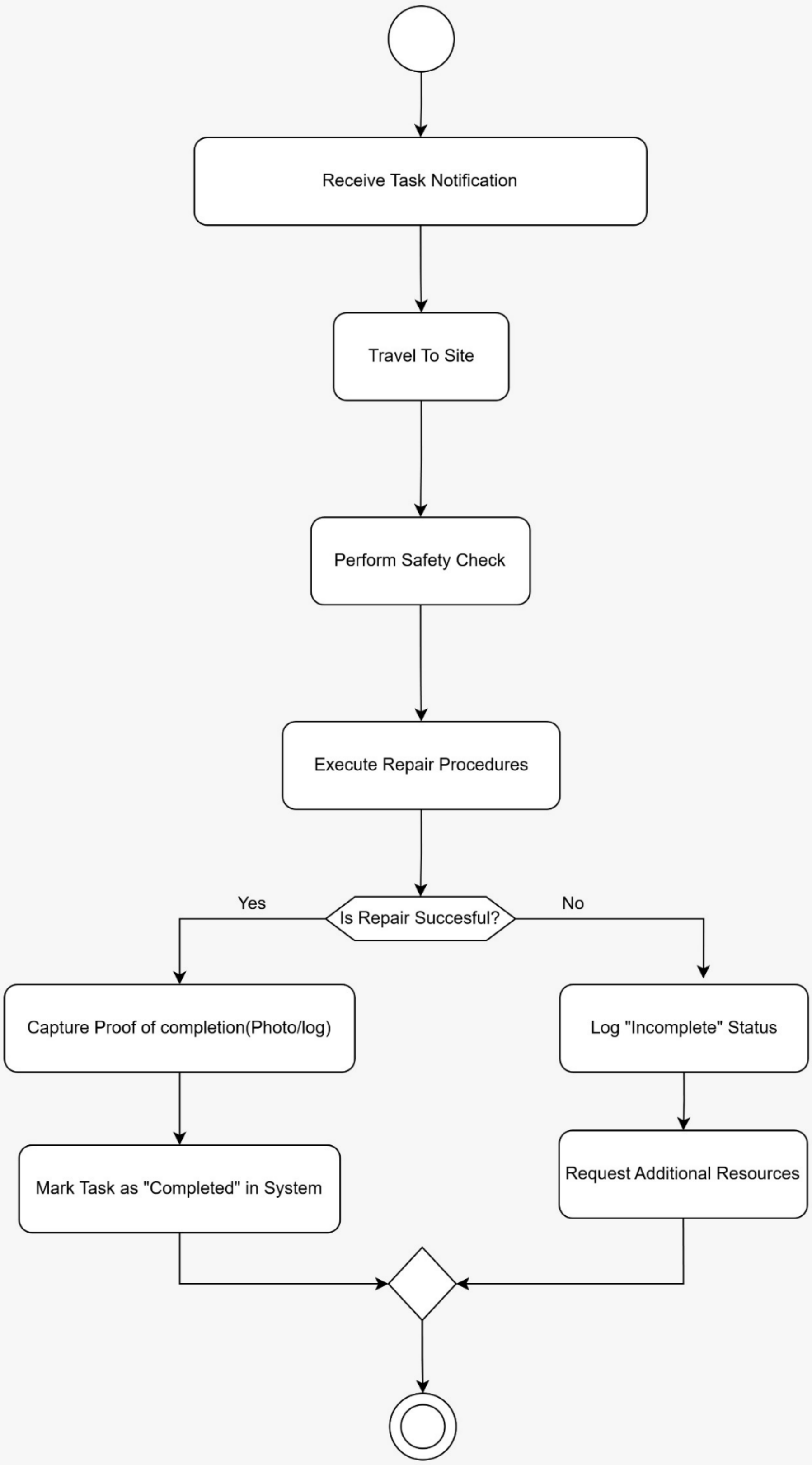
<u>Activity Diagram:(Department Officer) Analyze Infrastructure Issue</u>



Activity Diagram: (Department Officer) Schedule Repairs



Activity Diagram:(Field Engineer) Execute Maintenance Order



Sequence Diagram