DOKUZ EYLUL UNIVERSITY

ENGINEERING FACULTY

DEPARTMENT OF COMPUTER ENGINEERING

Prison Management System
Project Report

Emin Tekin 2023510185

Arda Şimşek 2023510211

Alaattin Yılmaz 2023510195

Prof. Dr. Semih Utku

IZMIR

07.05.2025

Table Of Contents

1. Abstract

This report presents the design and architectural blueprint of a software system aimed at managing prison. It outlines the system's overall structure, detailing the major components and modules and how they interact within the architecture. The document provides a class diagram that defines the system's primary classes, their attributes, and relationships. Additionally, it includes sequence diagrams to illustrate object interactions across key use cases, statechart diagrams to depict critical state transitions, and activity diagrams to model procedural workflows. The goal is to provide a clear understanding of how the system is structured and behaves, supporting both development and future maintenance efforts.

2. Introduction

2.1 Purpose of the System

The purpose of the Prison Management System is to streamline and digitize the administrative operations within conteinment facilities. It aims to improve the efficiency, accuracy, and security of tasks such as inmate registration, cell allocation, staff management, visitation scheduling, and tracking prisoner movements. The system is designed to reduce paperwork, ensure data consistency, and enhance overall facility management.

2.2 Scope

This system covers the core functionalities required for prison administration. It includes modules for inmate information management, sentence tracking, staff assignments, cell and block organization, visitor management, and security alerts. The system will be used by prison staff, administrators, and authorized personnel only. .

2.3 Overview of the Document

This document provides a comprehensive overview of the system's architecture, design, and behavior. It includes diagrams such as class diagrams, sequence diagrams for various use cases, statecharts for key components, and activity diagrams to illustrate workflows. Each section contributes to a complete understanding of how the system is structured and functions.

3. System Overview

3.1 System Architecture

The core of the system is a central object called System, which serves as the main access point for all operations. Users interact with the system exclusively through this object. The System object coordinates functionality by delegating responsibilities to three major manager components:

- PrisonerManager: Responsible for tracking and managing prisoner information, including registration, sentence details, movement history, and visitation records. Each prisoner is assigned to a specific Section within the prison.
- StaffManager: Manages all staff-related information such as accounts, roles, and assignments. Each staff member is associated with one or more Sections, representing the areas they supervise or work in.
- ReportManager: Handles the creation, retrieval, and storage of various reports, including prisoner activity, staff logs, and system summaries.

Another critical structure in the system is the Section class, which models the physical layout of the prison. Sections are organized in a tree-like hierarchy to represent blocks, rooms, or sectors. This hierarchical structure allows the system to search for available cells efficiently, add prisoners to appropriate areas, and model complex prison layouts.

Each Section can:

- Contain references to child Sections, forming a tree structure.
- Store prisoners currently assigned to it.
- Track assigned staff members.

This modular and centralized design ensures separation of concerns, where each manager focuses on a specific domain while the System object provides a unified interface for external interaction.

3.2 Overall System Design Description

In conclusion the system operates around a central controller object (System) that coordinates tasks by delegating them to specialized modules: PrisonerManager, StaffManager, and ReportManager. These managers handle domain-specific operations and interact with shared structures such as Section, which organizes the prison layout in a hierarchical tree. This design allows efficient management of prisoners, staff, and reports while keeping responsibilities modular and maintainable.

4. System Design

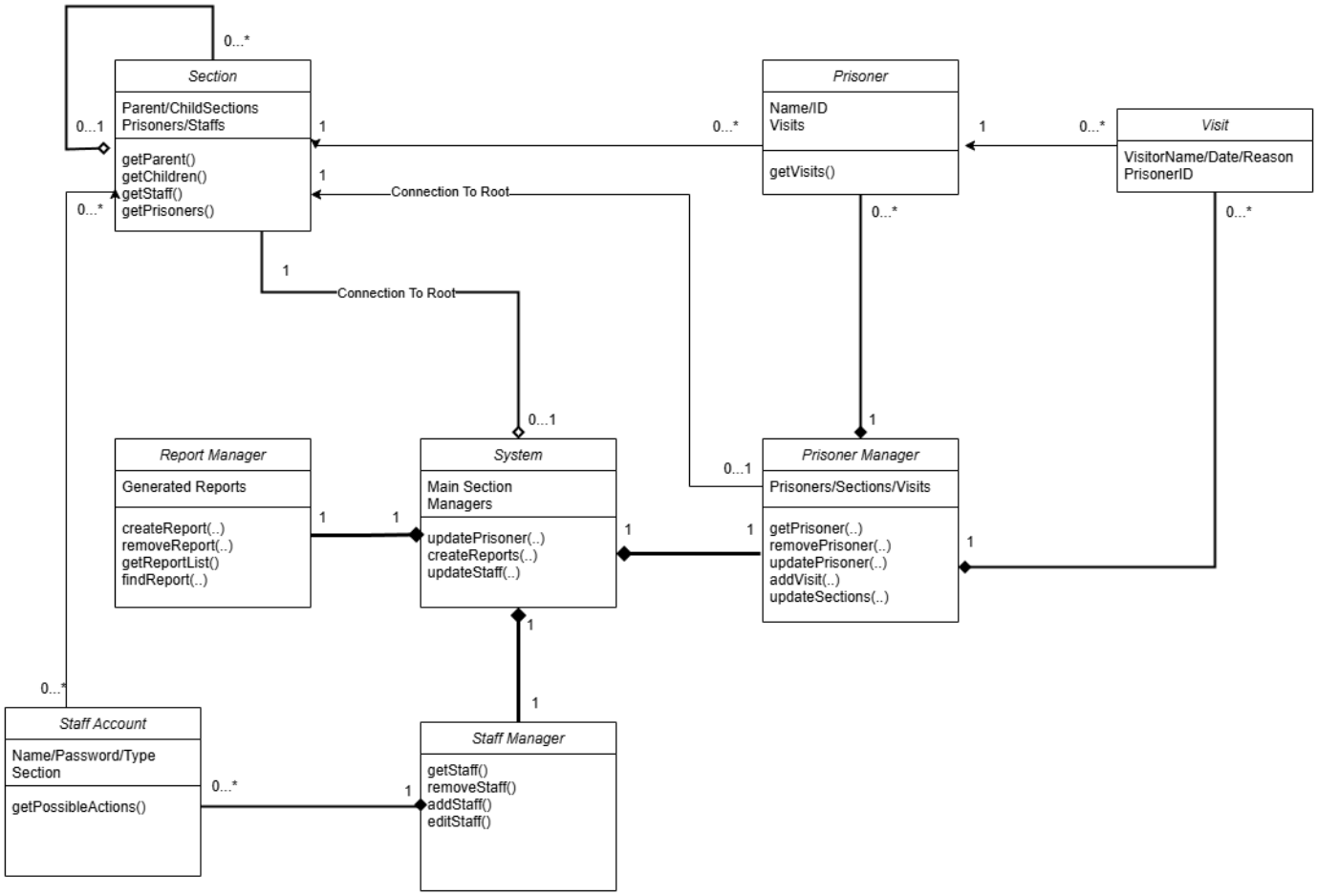
4.1 Components and Modules

The system is composed of several key components, each responsible for specific functional areas:

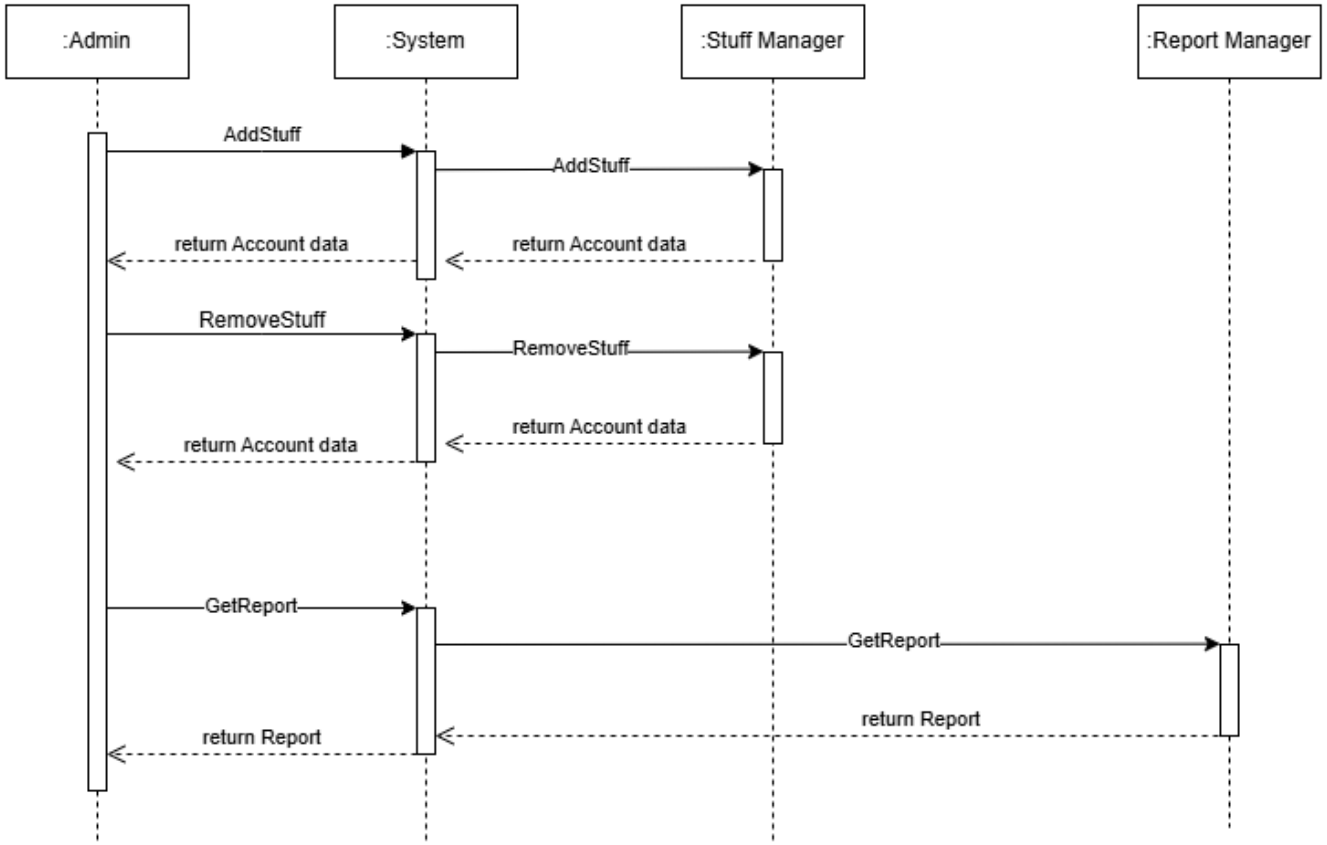
- System: The central controller that acts as the main access point for all user operations. It coordinates with other modules rather than handling logic itself.
- PrisonerManager: Responsible for storing, tracking, and updating prisoner data. It also manages prisoner visit logs and cell assignments.
- StaffManager: Manages all staff-related operations, including account management, authentication, and assigning staff to sections.
- ReportManager: Generates various system reports based on data from other modules. This includes prisoner summaries, staff activity, and system usage reports.
- Section: Represents the physical structure of the prison. Sections can form a tree structure to model complex prison layouts. Each section can contain prisoners and is associated with staff.
- Prisoner: A data entity representing an individual prisoner. Contains personal information and section assignment.
- Staff: A data entity representing a prison staff member, linked to one or more sections.
- Visit: Visit class referanced to the prisoner whom part of the visit and referanced from prisoner manager holds data like date, visitor name ect.

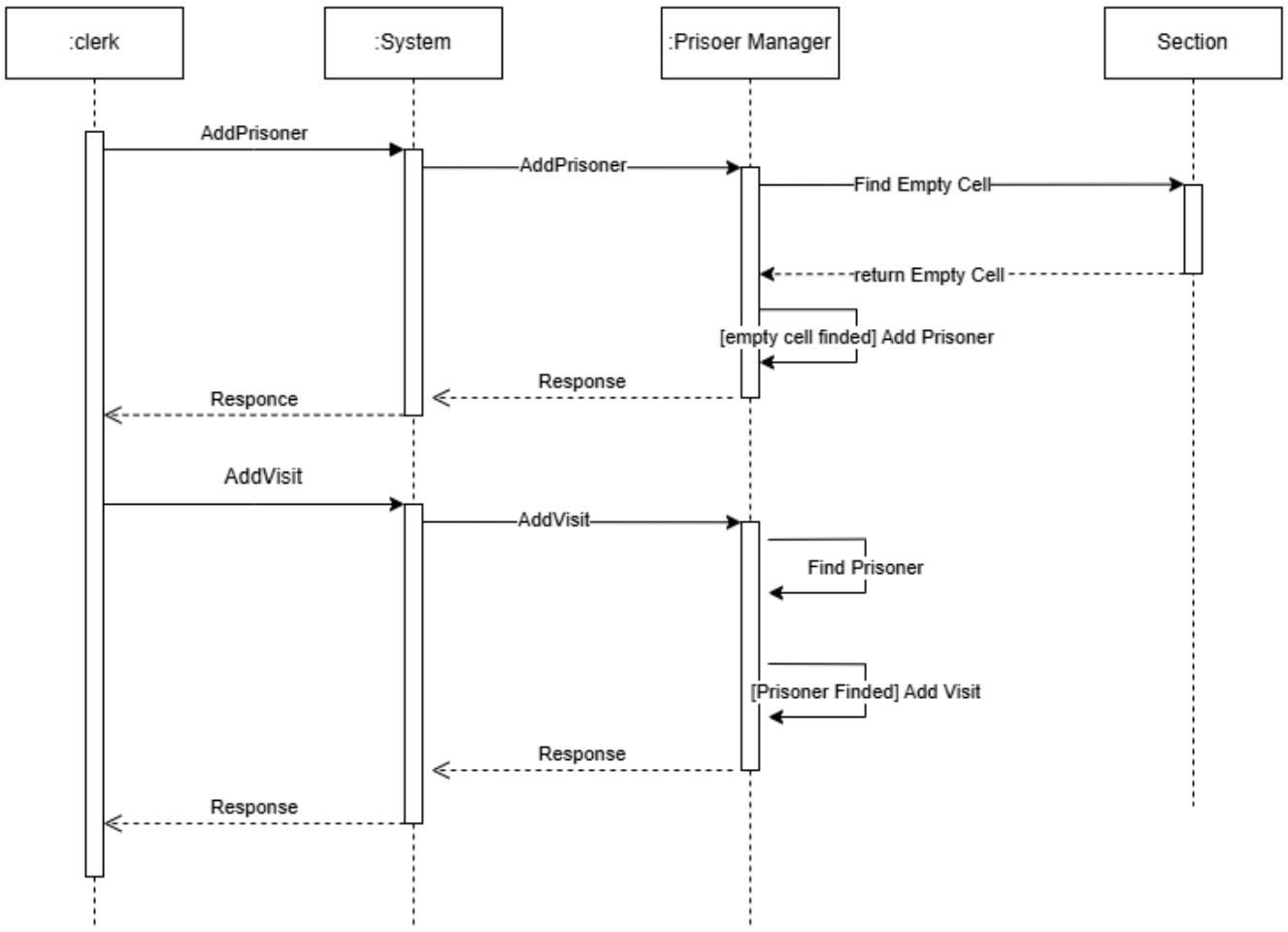
Each module is designed with clear responsibilities and minimal coupling to support maintainability and scalability.

4.2 Class Diagram



4.3 Sequence Diagrams : Add stuff, remove stuff, get report,add prisoner, add visit

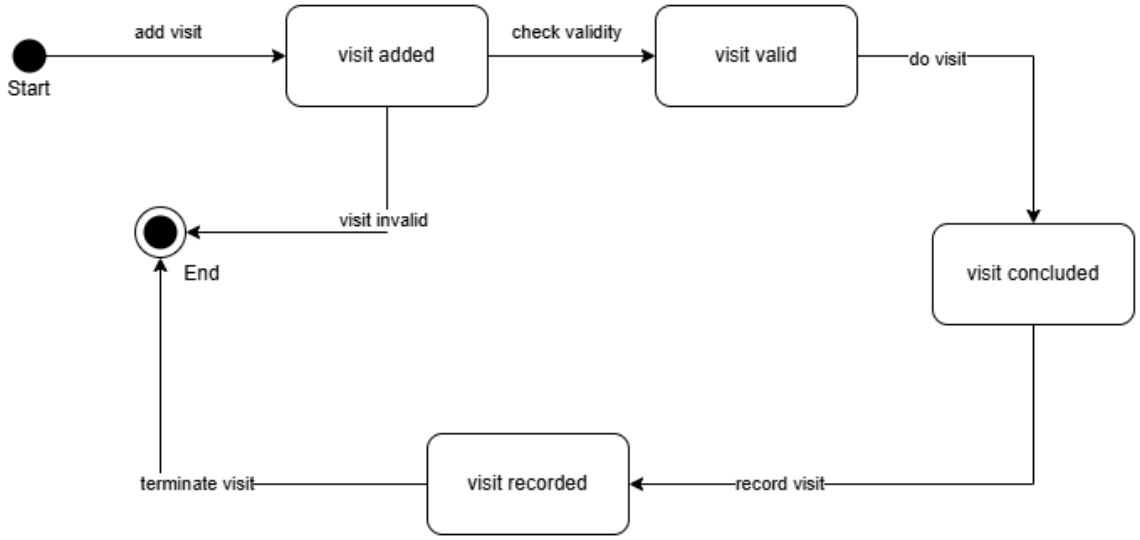

:Admin
:System
:Stuff Manager
:Report Manager
AddStuff
AddStuff
return Account data
return Account data
RemoveStuff
RemoveStuff
return Account data
return Account data
GetReport
GetReport
return Report
return Report

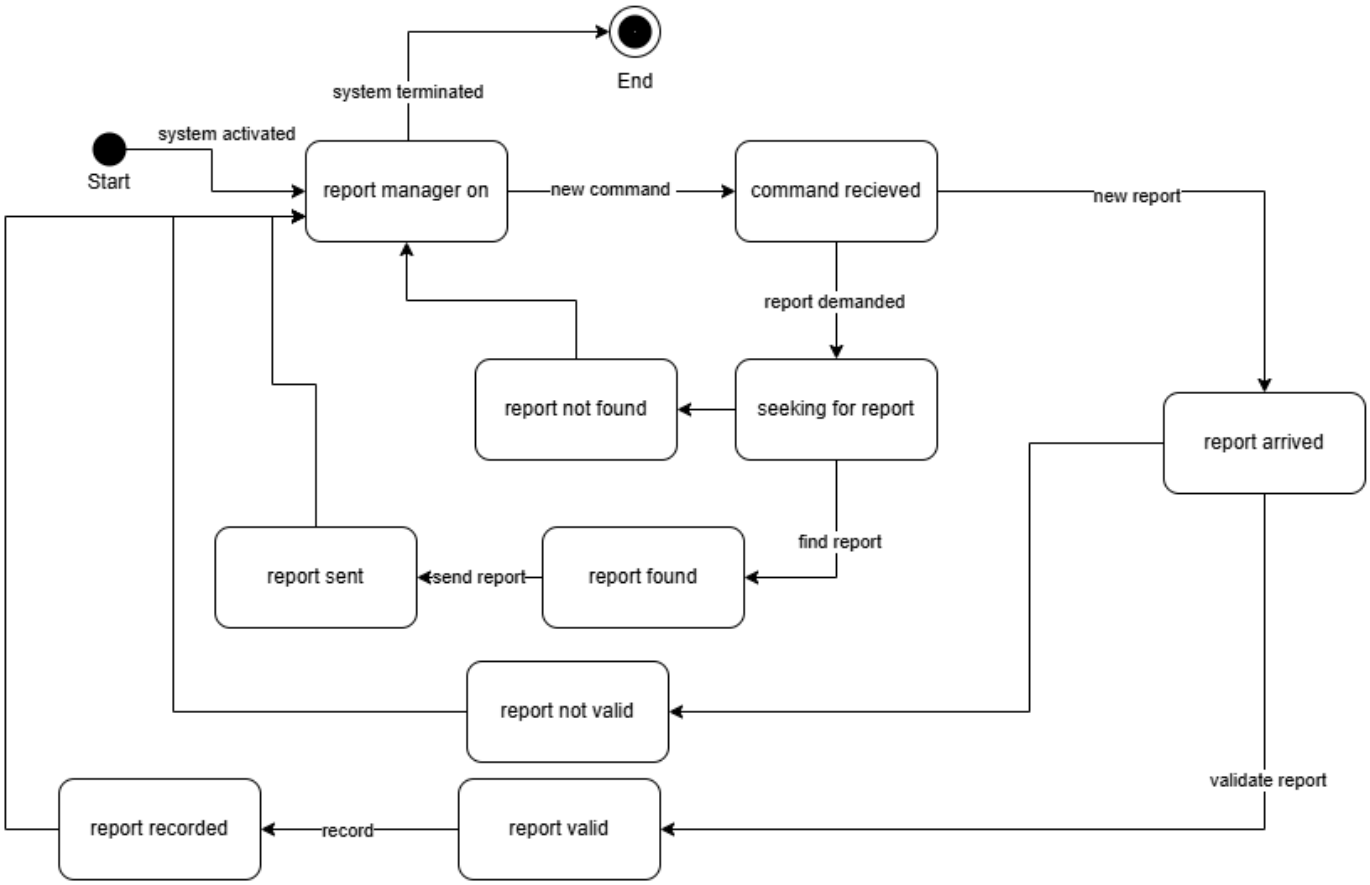

:clerk
:System
:Prisoer Manager
Section
AddPrisoner
AddPrisoner
Find Empty Cell
return Empty Cell
[empty cell finded] Add Prisoner
Response
Responce
AddVisit
AddVisit
Find Prisoner
[Prisoner Finded] Add Visit
Response
Response

5. Behavioral Models

5.1 Statechart Diagrams

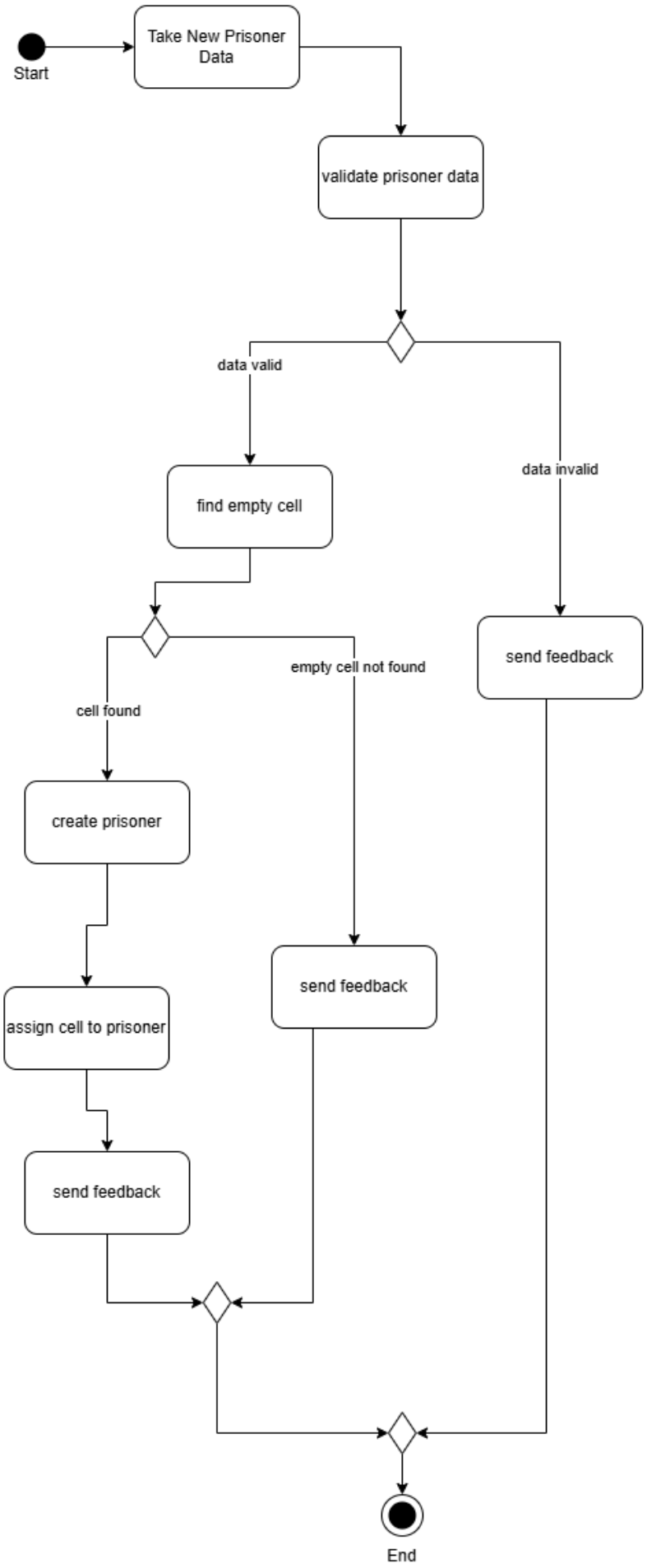
visit class ,report manager class



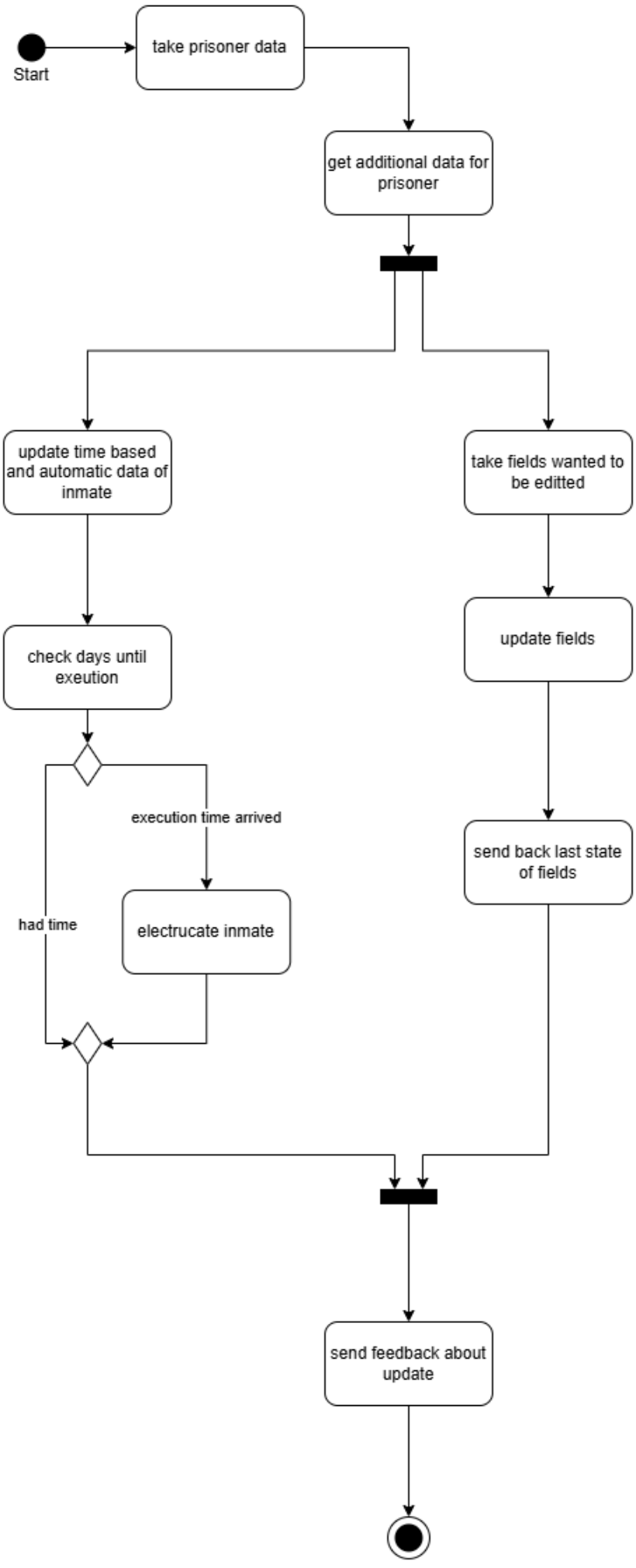


5.2 Activity Diagrams

Add New Prisoner



Update Prisoner Data



Register New Stuff Account