

Prison Management System

Nihar Niranjan S Nikith T Rajan Niranjay Ajayan Varun Raj R

Department of Computer Engineering
Model Engineering College
Thrikkakara, Kochi 682021

October 7, 2024

Table Of Contents

- 1 Introduction
- 2 SRS
 - Functional Requirements
 - Non-Functional Requirements
- 3 Architecture
 - Architecture Description
 - UML Diagrams
- 4 Result Screenshots
- 5 References

Introduction

The Prison Management System (PMS) provides a complete solution to modernize the administration of prisons in a state of the art manner. The traditional manual methods of record keeping and management used in correctional facilities can often cause errors, inefficiency, and security risks. PMS does away with the need for all these processes by moving them to a secured, centralized, and user-friendly platform and presenting significant improvements in accuracy, efficiency in day-to-day operations, and safety for the inmates and the prison staff.

Functional Requirements

Staff Management

- Login: Staff can securely log in using their staff ID and password.
- View/Manage Staff: Ability to view, register, update roles, or remove staff members.

Prisoner Management

- Add/Release Prisoners: Functions to add new prisoners and release them when required.
- Update Prisoner Information: Modify prisoner details, including reassignment to cells.
- Generate Prisoner Reports: Access comprehensive information on prisoners.

Functional Requirements

Cell and Task Management

- Allocate/Change Cells: Assign prisoners to cells or reassign them.
- Assign/Record Work: Assign jobs to prisoners and track their work hours

Visitor and Incident Tracking

- Register a new visitor and keep the visitors history
- Filter the records of prisoners by crime and generate statistics on the type of crime

Functional Requirements

- **Frontend:** The user interface is built using React and styled with CSS, making it responsive and easy to navigate. This ensures users can interact with the system smoothly, whether on desktop or mobile.
- **Backend:** Python powers the backend, managing all the logic and API communication to keep everything running efficiently behind the scenes.
- **Database:** MySQL is used to securely store all critical data, including prisoners, staff, and visitors.
- **Security:** For added safety, hashlib encrypts sensitive information, ensuring data integrity and secure access.

Non-Functional Requirements

Performance Requirements

- Handle access by multiple staff users
- Quick data retrieval

Security Requirements

- Ensure role-based access control for staff roles (admin, staff)
- Encrypt sensitive data (e.g., login credentials) using secure hashing techniques like hashlib.

Usability Requirements

- User-friendly React-based interface with smooth navigation.
- Mobile-responsive design using CSS

Architecture Description

The prison management system is a modular, multi-functional system divided into subsystems that manage various aspects of the prison, such as staff, prisoners, cells, tasks, visitors, and crime tracking. Each subsystem handles a specific set of responsibilities, allowing for seamless coordination between them to achieve the overall functionality.[S+24]

Use-Case Diagram

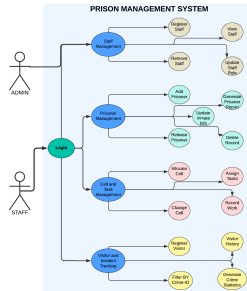


Figure: Use-Case Diagram

Class Diagram

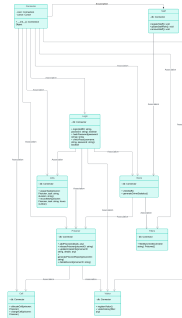


Figure: Class Diagram

Activity Diagram

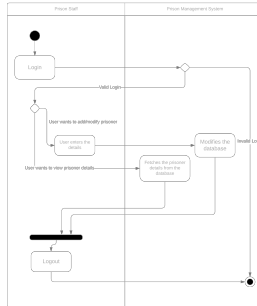


Figure: Activity Diagram

Sequence Diagram

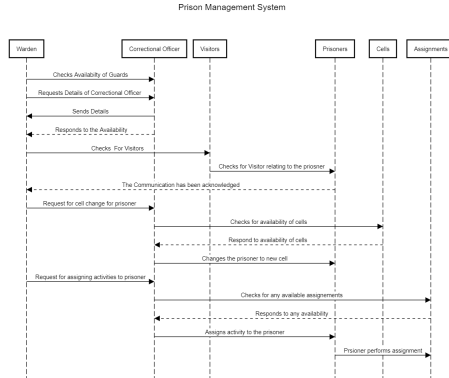


Figure: Sequence Diagram

ER Diagram

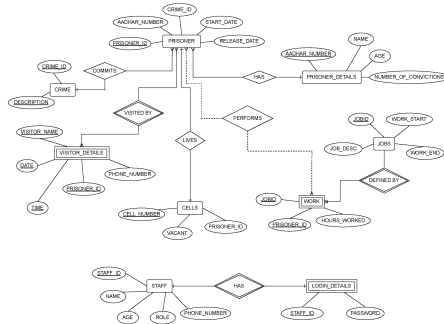


Figure: ER Diagram

Screenshots

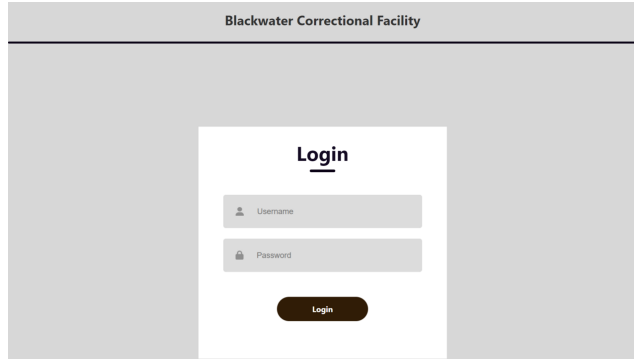


Figure: Login Screen

Screenshots

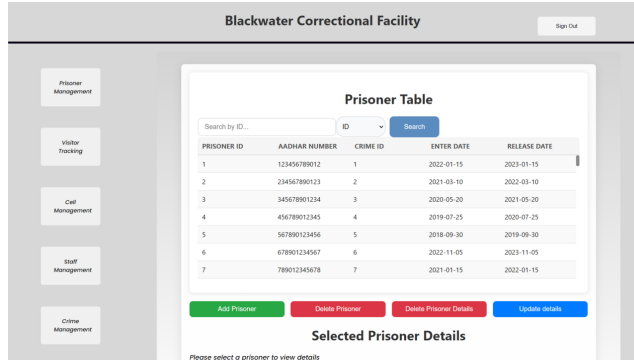


Figure: Prisoner Management

Screenshots

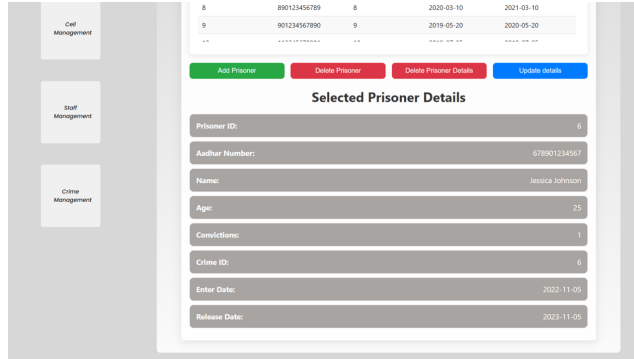


Figure: Prisoner Details

Screenshots

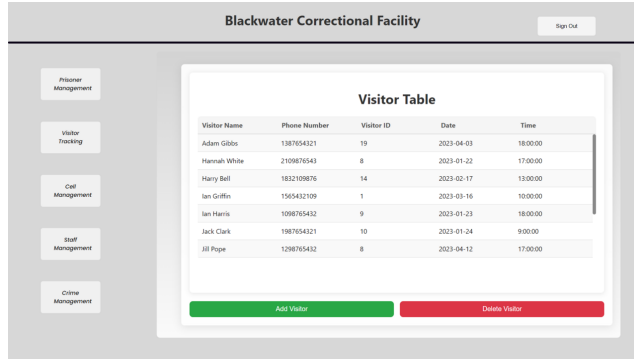
The screenshot displays the 'Blackwater Correctional Facility' interface. On the left, a sidebar contains navigation buttons: 'Prisoner Management', 'Visitor Tracking', 'Cell Management', 'Staff Management', and 'Crime Management'. The main area is divided into a top section with a 'Sign Out' button and a bottom section titled 'Selected Prisoner Details' with a message 'Please select a prisoner to view details'. A modal dialog titled 'Add Prisoner' is open in the center, featuring a close button (X) and several input fields: 'Name' (with a placeholder 'Enter Name'), 'Age' (with a placeholder 'Enter Age'), 'Aadhar No.' (with a placeholder 'Enter Aadhar No.'), 'Crime ID' (with a placeholder 'Enter Crime ID'), 'Entry Date' (with a date picker showing 'dd-mm-yyyy'), and 'Release Date' (with a date picker showing 'dd-mm-yyyy'). Below the inputs is an 'Add' button. In the background, a table lists prisoners with columns for 'ENTER DATE' and 'RELEASE DATE'. The table contains the following data:

| | ENTER DATE | RELEASE DATE |
|---|------------|--------------|
| 1 | 2022-01-15 | 2023-01-15 |
| 2 | 2021-03-10 | 2022-03-10 |
| 3 | 2020-05-20 | 2021-05-20 |
| 4 | 2019-07-25 | 2020-07-25 |
| 5 | 2018-09-30 | 2019-09-30 |
| 6 | 2022-11-05 | 2023-11-05 |
| 7 | 2021-01-15 | 2022-01-15 |

At the bottom of the main area, there are four buttons: 'Add Prisoner' (green), 'Delete Prisoner' (red), 'Delete Prisoner Details' (red), and 'Update details' (blue).

Figure: Add Prisoner Dialog

Screenshots



Blackwater Correctional Facility Sign Out

Visitor Table

| Visitor Name | Phone Number | Visitor ID | Date | Time |
|--------------|--------------|------------|------------|----------|
| Adam Gibbs | 1387654321 | 19 | 2023-04-03 | 18:00:00 |
| Hannah White | 2109876543 | 8 | 2023-01-22 | 17:00:00 |
| Harry Bell | 1832109876 | 14 | 2023-02-17 | 13:00:00 |
| Ian Griffin | 1565432109 | 1 | 2023-03-16 | 10:00:00 |
| Ian Harris | 1098765432 | 9 | 2023-01-23 | 18:00:00 |
| Jack Clark | 1987654321 | 10 | 2023-01-24 | 9:00:00 |
| Jill Pope | 1298765432 | 8 | 2023-04-12 | 17:00:00 |

Add Visitor Delete Visitor

Figure: Visitor Management

Screenshots

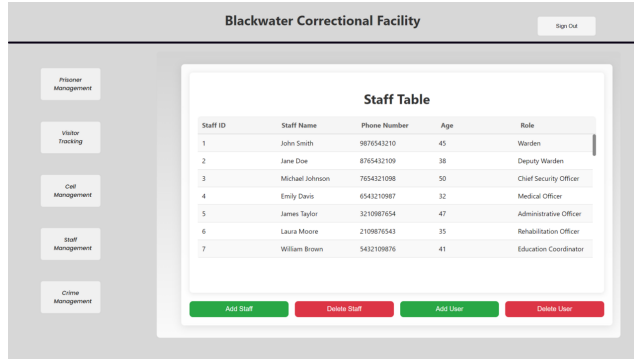


Figure: Staff Management

Screenshots

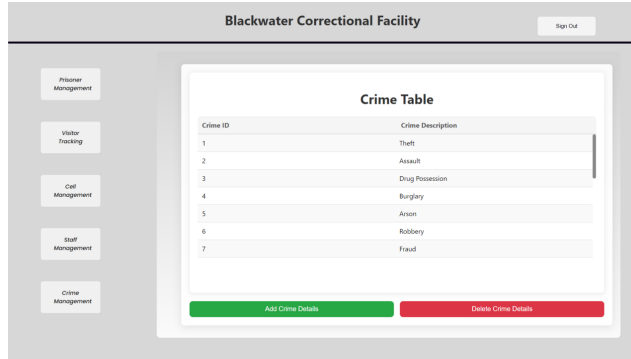


Figure: Crime Management

References

[S+24] Nihar Niranjan S et al. "Prison Management System Software Design Document". [Provided in the repo. 2024.](#)