Detention Admin Website

Mario Guriuc, Vlad Spiridon

I. Introduction

A. Purpose

The purpose of this document is to outline the requirements for the development of a web application for managing visits received by individuals sentenced to serve time in a correctional facility, whether it be a prison or a correctional home. The application will provide functionality to record visitor details, visit information, and generate statistical reports for analysis.

B. Scope

The application will allow authorized personnel to record and manage visitation data efficiently. It will facilitate the entry and retrieval of information related to visitors, including their identity, photograph, relationship with the inmate, nature of the visit, and associated details. Additionally, the application will generate statistical reports to analyze visitation trends and patterns over time.

C. Definitions, Acronyms, and Abbreviations

API: Application Programming Interface
CSV: Comma-Separated Values
JSON: JavaScript Object Notation
HTML: HyperText Markup Language
REST: Representational State Transfer
GraphQL: Graph Query Language

D. Overview

This document will cover the functional and non-functional requirements of the web application, including user roles, system features, data management, security measures, and interfaces. It aims to provide a comprehensive guide for the development team to implement the system effectively.

II. Overall Description

A. Product Perspective

The web application will function as a standalone system, providing a user-friendly interface for managing visitation data.

B. Product Features

The key features of the application include:

Visitor Management: Recording visitor details including identity, photograph, relationship with the inmate.
Visit Information: Capturing data such as visit date, duration, nature of visit, items exchanged, discussions summary, inmate's health and mood, witnesses, etc.
Statistical Reporting: Generating reports in HTML, CSV, and JSON formats based on various criteria such as individual visits, time periods, types of offenses, etc.
API Access: Providing essential functionalities through a REST/GraphQL API for seamless integration with other systems or external applications.

C. User Classes and Characteristics

The primary users of the system include:

Prison Administrators: Responsible for overall management and supervision of the application.
Correctional Officers: Enter visitation data and generate reports.
Legal Advisors: Access visitation records for legal purposes.
Visitors: Provide information about themselves and their relationship with the inmate before scheduling a visit.

D. Operating Environment

The application will be web-based and accessible through modern web browsers such as Google Chrome, Mozilla Firefox, and Safari. It will be hosted on a secure server infrastructure compliant with institutional security standards.

E. Design and Implementation Constraints

The system should adhere to relevant legal regulations regarding data privacy and confidentiality.

F. Assumptions and Dependencies

It is assumed that users have basic computer literacy and internet access. The system depends on reliable network connectivity for real-time data entry and retrieval.

III. System Features

A. Visitor Management

1) Description: This feature allows users to manage visitor information, including identity, photograph, and relationship with the inmate.

2) Inputs:

Visitor details: Name, contact information, photograph, relationship with the inmate.
Inmate details: Identification number, name, facility location.

3) Outputs:

Visitor records stored in the database for future reference.
Ability to associate visitors with specific inmates.

B. Visit Information

1) Description: This feature enables users to record detailed information about each visit, including date, duration, nature of visit, items exchanged, discussions summary, inmate's health and mood, witnesses, etc.

2) Inputs:

Visit date and time.
Duration of visit.
Nature of visit (e.g., legal consultation, family visit).
Items exchanged between visitor and inmate.
Summary of discussions.
Inmate's health and mood assessment.
Witnesses present during the visit.

3) Outputs:

Visitation records stored in the database.
Ability to generate reports based on visitation data.

C. Statistical Reporting

1) Description: This feature allows users to generate statistical reports analyzing visitation trends and patterns over time.

2) Inputs:

Criteria for report generation (e.g., individual visits, time periods, types of offenses).
Output format (HTML, CSV, JSON).

3) Outputs:

Statistical reports presented in the specified format.
Insights into visitation patterns for decision-making purposes.

D. API Access

1) Description: This feature provides essential functionalities through a REST/GraphQL API for seamless integration with other systems or external applications.

2) Inputs:

API requests for data retrieval, insertion, or modification.
Authentication credentials for authorized access.

3) Outputs:

Responses to API requests containing requested data or confirmation of successful operations.

IV. External Interface Requirements

A. User Interfaces

The application will feature intuitive user interfaces for data entry, retrieval, and report generation. It will include forms, tables, and interactive elements for ease of use.

B. Hardware Interfaces

The system will be accessible via standard computing devices with internet connectivity, including desktop computers, laptops, tablets, and smartphones.

C. Software Interfaces

The application may interface with external systems for authentication purposes or data integration. It should be compatible with commonly used web browsers and operating systems.

D. Communication Interfaces

The system will communicate with users through web-based interfaces and may utilize email notifications for important updates or reminders.

V. Backend and Database

The backend of the application will be developed using PHP, a server-side scripting language known for its versatility and ease of integration. PHP will handle the logic of the application, including data processing, authentication, and communication with the database.

For the database management system, MongoDB will be utilized. MongoDB is a NoSQL database that offers flexibility and scalability, making it suitable for storing and managing the diverse data structures associated with visitation records. It provides features such as document-based storage, dynamic schemas, and high availability, which are advantageous for the requirements of the detention administration website.