

**Title: Prisoner Management System (SRS)**

**Course Code:** SEC-2070

**Course Title:** Software Construction and Management

**Department:** Department of Computer Science

**Section:** BS Software Engineering (2019-21)

**Semester:** 5th

Submitted By:

**Abdullah Anas (19-NTU-CS-1086)**

**Mudassar Ali (19-NTU-CS-1104)**

**Touqeer ul Hussnain (19-NTU-CS-1128)**

Submitted to:

**Dr. Muhammad Asif Ayub Sb.**

**National Textile University Faisalabad**

1. **Table of Contents:**

1 Table of Contents ...........................................................................................................................................................2

2 Problem Statement ....................................................................................................................................................... 4

3 Solution to Problem ...................................................................................................................................................... 4

3.1 Functional Requirements ........................................................................................................................... 4

3.2 Non-Functional Requirements ................................................................................................................... 4

4 Introduction ...................................................................................................................................................................5

4.1 Background ................................................................................................................................................ 5

4.2 Investigation & Analysis Methodology........................................................................................................5

4.2.1 System Investigation ............................................................................................................... 5

4.2.2 Analysis Methodology ............................................................................................................. 5

4.2.2.1 Scope and Limitations........................................................................................... 5

4.2.3 Unified Modelling Language (UML)..........................................................................................5

4.2.3.1 Use Case Diagram…………………………………...............................................................6

4.2.3.2 Sequence Diagram .................................................................................................7

4.2.3.2.1 Sequence for Prisoner handling……... ……………………………..……….........7

4.2.3.2.2 Sequence for handling Guard data ......................................................8

4.2.3.2.3 Sequence for handling Jailers Data......................................................9

4.2.3.2.3 Sequence for handling Food Manager................................................10

4.2.3.2.4 Sequence for Notification………………...................................................11

5 Constraints...................................................................................................................................................................…12

5.1 Scalability.....................................................................................................................................................12

5.2 Data and Function Mapping ........................................................................................................................12

5.3 Proprietary hardware and software............................................................................................................ 12

5.4 Batch updates vs. (close) Real-time updates .............................................................................................. 12

5.5 Project Schedule ......................................................................................................................................... 12

6 Operational Requirements ............................................................................................................................................ 12

6.1 Application Services and Technical support ............................................................................................... 12

6.2 Administration Features ............................................................................................................................. 12

6.3 System Interface ………………………………........................................................................................................ 12

6.4 System hardware fail over and routine back up.......................................................................................... 12

7 Process Requirements.....................................................................................................................................................13

7.1 Database Transaction...................................................................................................................................13

7.2 Data integrity................................................................................................................................................13

7.3 Data validation .............................................................................................................................................13

7.4 Performance ................................................................................................................................................13

7.5 Data repository ...........................................................................................................................................13

7.5.1 Class view .................................................................................................................................13

8 Output Requirements.....................................................................................................................................................14

8.1 View Data…………………………………………… .....................................................................................................14

8.2 Prisoner Reports and summaries .........................................................................................................…….14

9 Hardware Requirements ...............................................................................................................................................14

9.1 Client Computers.........................................................................................................................................14

9.2 Production and Support System..................................................................................................................14

10 Software Requirements ...............................................................................................................................................14

10.1 Client Operating Systems ..........................................................................................................................14

10.2 Mainframe system ....................................................................................................................................14

10.3 Licenses......................................................................................................................................................14

1. **Problem Statement:**

In manual Prisoner management system Jailers are working on manual system to keep the record. This type of system is inefficient. It costs too much efforts to maintain prisoner’s record. It is also effected by poor management. There are also risks of data loss due to paper work. It is also quite difficult to manage data in hard form. It is not easy to change or update some data. There is also storage issue for a long record for future purposes. So, the user wants to automate the system to avoid all difficulties.

1. **Solution to Problem:**

To avoid from all the difficulties defined above we are going to automate the system named as Prison management system. This system is made to keep records about the prisoners, jailers, guards and food manager. Admin and jailer can log in the system.

Admin can maintain the records of Prisoners, Jailers, Guards, and Food Manager.Jailers only deal with the entities of Prisoners and Guards. Jailers can change attributes like time-shift duty hours of guards.

This system talks about when any prisoner gets their punishment period over so that he/she can be released. There is a Notice board to show notice about the release of prisoner.

The prison management system can be implemented in every prison without any problem. This system has the capability to maintain an infinite number of records. It is very useful as the written papers have a limited time period and can get lost but in the prison management system, this is not possible as a backup file will be created when any change is made in any record.

There is no option to delete a prisoner’s record because it may be required later by the government to know any details about the person and can help in the tracking of the prisoner.

**3.1 Functional Requirement:**

The main function of the system is to store, update and maintain Prisoner’s data (i.e. Prisoner ID, Name, Crime date, Age, Crime, Punishment, Release Date), Jailer data (i.e. Jailer ID, Name, Phone Number, Age, Email, Address, Password), Guard data (i.e.ID, Name, Phone Number, Duty time, Shift time), and Food Manager (i.e. ID, Name, Experience, Address, Phone Number, Age, Email).

Admin has access to change and update Prisoner, Jailer, Guards, and Food Manager. Jailer has access to change and maintain only Prisoner’s data and Guard data. Food Manager has the duty to check and maintain daily food schedule. Similarly, system has the Notice board to show Release date of Prisoners. This system is also accessible for Government for their uses.

**3.2 Non-Functional Requirement:**

System should store record of minimum 500 prisoners. System should be applicable in any Window version. System should secure data access (only valid user can access the data). Users of system should find every functionality on front page.

1. **INTRODUCTION:**
   1. **Background:**

The background of the system is that user faces too much difficulties in maintaining and storing the data manually. It is a time requiring process. It requires too much hard work and is less efficient. There are also chances of data loss. Therefore, to avoid all these ambiguities and difficulties we are going to automate the system.

* 1. **Investigation & Analysis Methodology:**
     1. **System Investigation:**

The Prisoner Management System holds the record of prisoners using MS SQL. The data of prisoners, Guards is stored in MS SQL repository. Jailer and Admin can view, update, and delete the stored record through given functions.

* + 1. **Analysis Methodology:**
       1. **Scope and Limitations:**

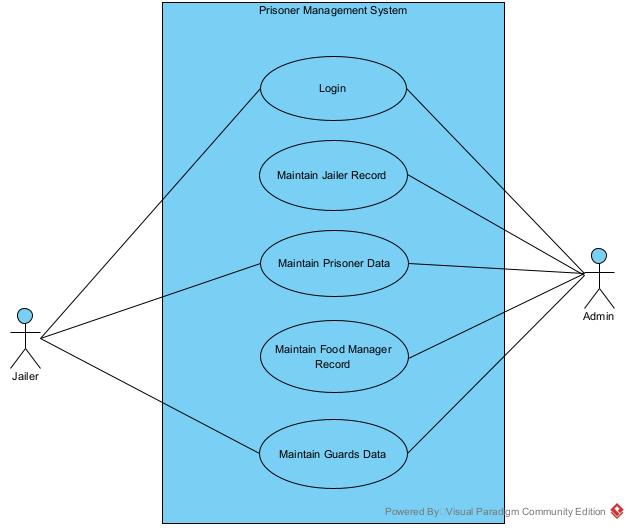
Analysis methodology will involve requirement analysis, data analysis, process analysis, and application architecture:

* Requirement analysis: System I/O description, user requirement definition, functional and security requirement
* Data analysis: Involve data collection process, data validation, data storage, manipulation and retrieval
* Process analysis: Data/process flow analysis, process decomposition and system interfaces
* Application architecture: Analyse application information structure, usability, user interface design, interaction and application implementation.
  + 1. **Unified Modelling Language (UML):**

UML will be used again for the graphical representation and documentation of the design. The system will primarily concern itself with the prisoner’s management. Jailer will login in the software. It will show records, manage guards, add prisoners.

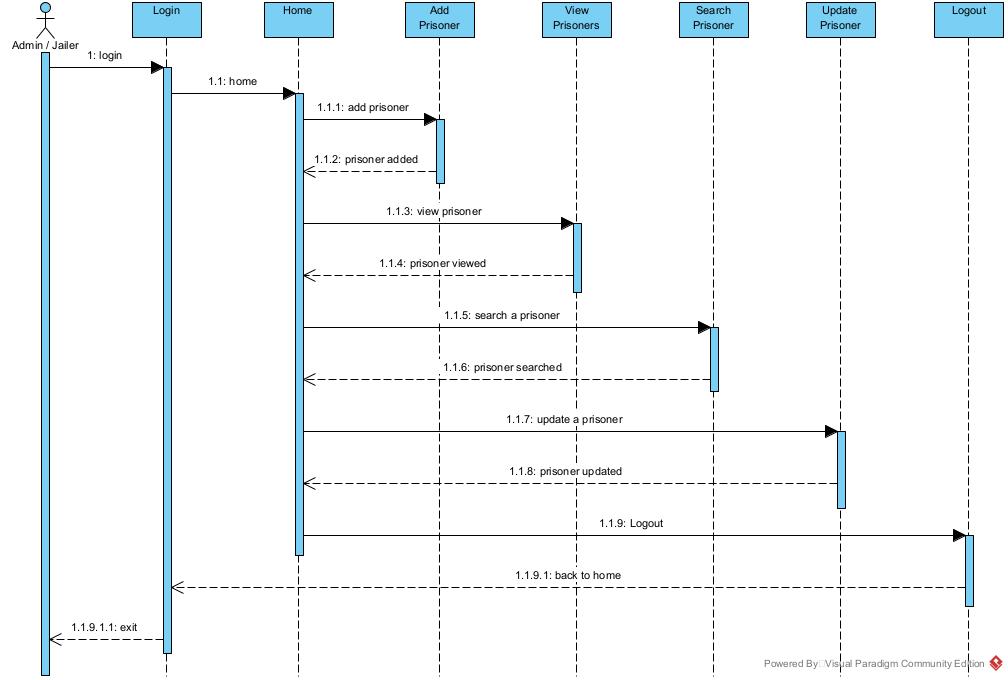
**Prisoner Management System**

* + - 1. **Use Case Diagram:**

****

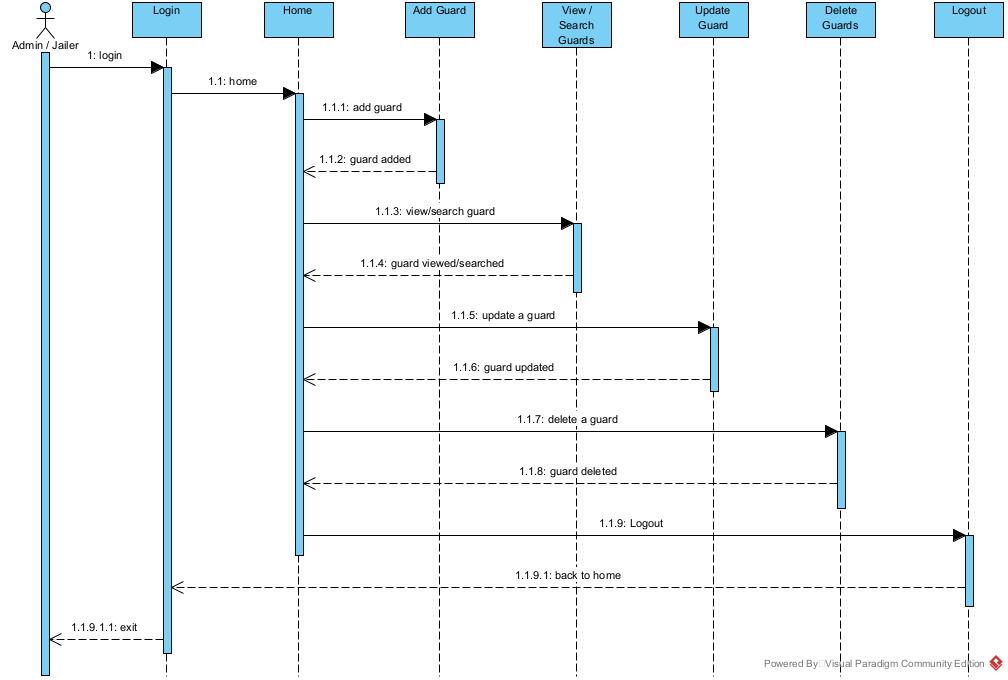
In Use-Case Diagram we simply show the actors and the functionality they are performing. Functionalities of Actors are shown in the form of use cases. Here, we have 2 actors i.e. Admin, and Jailer.

* + - 1. **Sequence Diagram:** A **sequence diagram** shows the sequence of messages passed between objects. Sequence diagrams can also show the control structures between objects.
         1. **Sequence for Prisoner Handling**

****

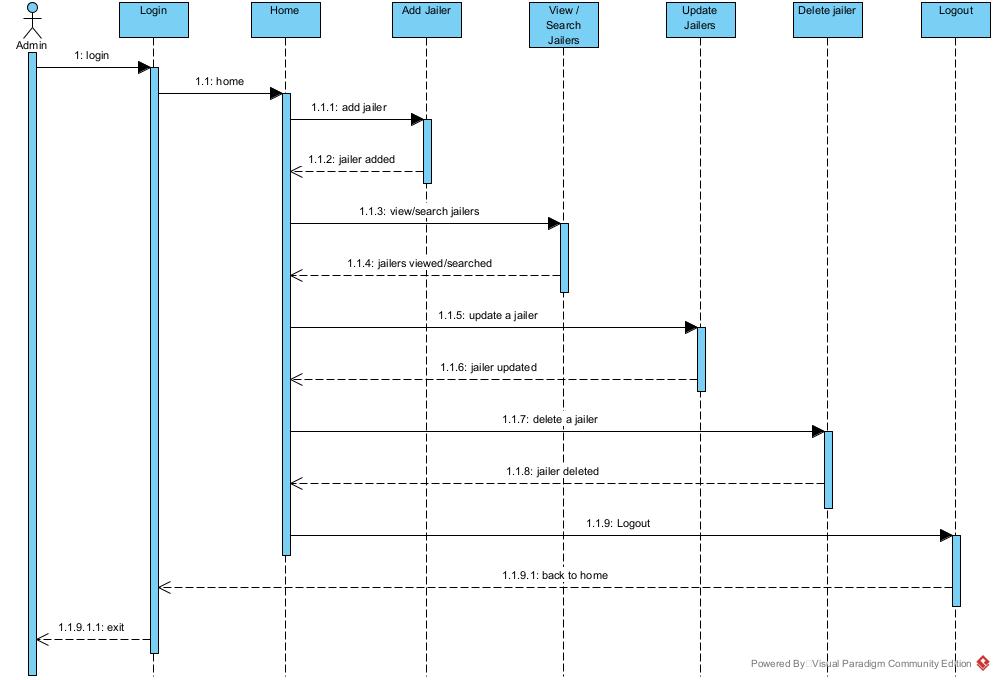
**Sequence Diagram for Prisoners Data Handling.**

* + - * 1. **Sequence for Handling Guard data**

****

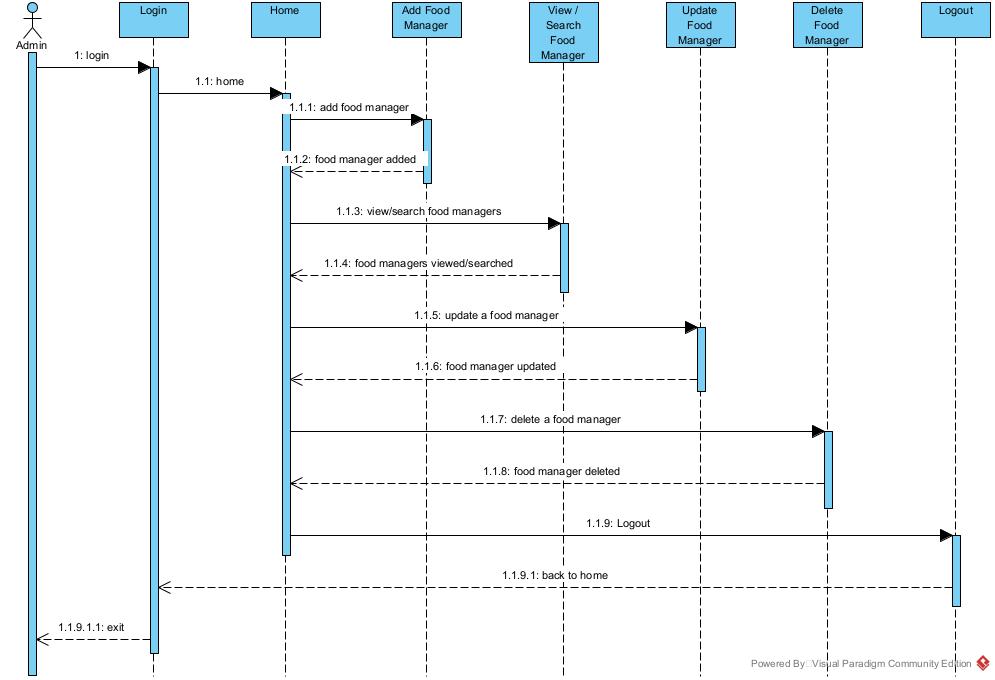
**Sequence Diagram for Guards Data Management**

* + - * 1. **Sequence for Handling Jailer Data**

****

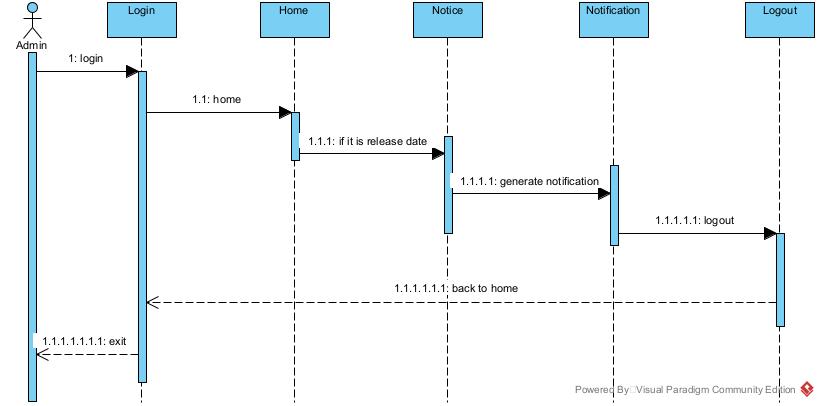
**Sequence Diagram for Jailers Data Handling.**

* + - * 1. **Sequence for handling Food Manager**

****

**Sequence Diagram for Food Manager’s Data Management.**

* + - * 1. **Sequence for Notification**

****

**Sequence Diagram for Notification.**

1. **Constraints:**
   1. **Scalability:**

The PM system does not scale well to increasing system demands. PMS’s underlying operating system was not designed to handle and resolve concurrent transactions. Error handling is also limited to few anticipated or common errors.

* 1. **Data and Function Mapping:**

A new function added to the mainframe-based Prisoner Management system cannot be readily mapped to the existing PM system. For example, a new Jailer added to the mainframe-based Prisoner Management system will require a source code change and recompilation of the main PMS program.

* 1. **Proprietary hardware and software:**

PM system requires proprietary hardware and software from modern Technology in order to be operational.

* 1. **Batch updates vs. (close) Real-time updates:**

There is no real-time update of Prisoner Management system.

* 1. **Project Schedule:**

There is almost 3-month time frame to implement a production system of an PM system from project. We’ll deploy it till the end of 5th semester.

1. **Operational Requirements:**
   1. **Application Services and Technical support:**

Programmers and application developers will have access to source code to address bugs or system enhancements as deemed necessary.

* 1. **Administration Features:**

System security and access levels are provided in the online system. There are varying levels of system access and functional authority. Only authorized system Jailer has access to all Prisoner’s and Guard’s record.

* 1. **System Interface:**

The Jailer and Admin has access to all its functionalities through different buttons. Different screens will be shown for related activities. Jailer will interact through different buttons with system.

* 1. **System hardware fail over and routine back up:**

Computer operations centre will handle system hardware tasks such as hardware maintenance, fail over, scheduled system patches and maintenance.

1. **Process Requirements :**
   1. **Database transaction:**

The system must be able to send, receive and trigger transaction to SQL Server registration database system.

* 1. **Data integrity:**

Only authorized jailer can access all types of data regarding database.

* 1. **Data validation:**

System will fetch exactly the same as required data. Data error from the user’s end and from the back-end database-processing end must be gracefully handled. There will be data validation as part of the Prisoner Management System.

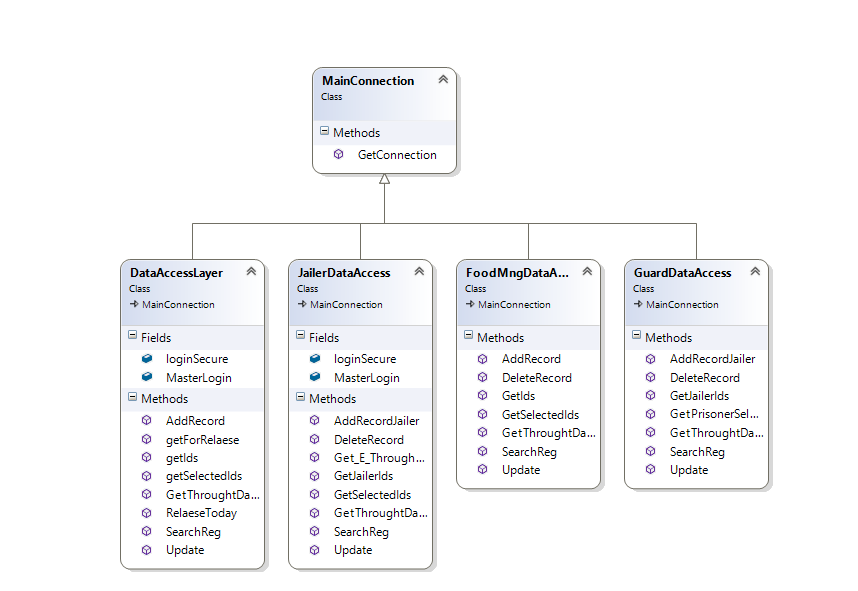
* 1. **Performance:**

The maximum wait over a query will be less than 3 seconds. System will respond within this time.

* 1. **Data repository:**

The PM system will maintain the SQL Server management system will be used as the main repository of data.

* + 1. **Class View :**



1. **Output Requirements:**
   1. **View Data:**

View data command will allow us to see data of prisoners and guards stored in SQL Database. We can search prisoner data at any time. System will allow us to generate a backup file. System will also generate a notification on release date of a prisoner.

* 1. **Prisoner Reports and summaries:**

Government and Security institutes must be able to extract summarized and entered data into meaningful information. All records will be archived but accessible on demand.

1. **Hardware Requirements:**
   1. **Client Computers:**

Windows client computers

* 1. **Production and support systems:**

Server computer(s) and related hardware support (redundant drives, UPS, etc.)

1. **Software Requirements:**
   1. **Client Operating Systems**

* Windows
  1. **Mainframe system**
* SQL Server Database
  1. **Licenses**

Valid licenses are required to run software from third party vendors:

* To use application development tools
* To use application program and database software in development, test and production mode.