**­­Software Requirement Specification (SRS) for Criminal Record Management (CRM)**

**1. Introduction**

Criminal Record Management (CRM) is a web based application. This software provides facility for reporting crimes, complaints, missing persons, show most wanted person details online. Any number of clients can connect to the Server.

* 1. **Purpose:**

This Software Requirements Specification provides a complete description of all the functions and specifications of the Criminal Record Management System (CRMS). Criminal Record Management System (CRMS) allows user to store police department’s case details, Complaint Details, FIR Details, etc. This Software Package allows Police Departments to store all the details related to the department and use them whenever necessary. This project will also be able to provide reports of various cases, FIR report, charge sheet report, Most Wanted Criminals record, payroll, attendance reports and also be able to upload and view criminal photos and scanned documents.

* 1. **Scope:**

**In Scope:**

1. The system should have a login.

2. System should support for Data Entry module for Complaint registration for each prisoner entering in the Jail.

3. Police officers should have a read only access to the information of prisoners and view complaint status.

4. Criminal register management.

5. The user should be able to view all most wanted persons which can be given by the administrator.

6. Data managers must be able to add or update the data of Jailers.

7. User authentication.

8. Maintenance of database about criminals and their details of crime.

**Out of Scope:**

1. FIR Management

2. Manage list about existed crimes

**1.3 Definitions, Acronyms, and Abbreviations:**

**Acronyms and Abbreviations:**

SRS – Software Requirements Specification

CRMS – Crime Record Management System

Subjective satisfaction – The overall satisfaction of the system

End users – The people who will be actually using the system

HTML – Hyper Text Mark-up Language

GUI- Graphical User Interface

**Definitions:**

* **Criminal**-criminals are the persons whose guilt/offence are proven in court and given punishment by court.
* **Most-wanted**-These Criminals are still not arrested or they flee away from police custody and police are searching for them.
* **Released person**-They are released by court as the charge against them not proved.
* **Investigation Officer-**Investigation officer is the police officer who investigates the case. There may be more than one investigation officers in one police station.

**1.4 References:**

Appendix A: IEEE SRS Format

Appendix B: <http://www.apit.ap.gov/>

Appendix C: <http://www.appstatepolice.org/>

**1.5 Overview:**

Criminal Record Management is a web based application. This application provides facility for reporting crimes, complaints, show most wanted person details online. Jail Superintendent is allotted highest authority of ministry. The duty of the Jail superintendent is to register the Criminal and update the records. Section 2 gives an overall description of the software. It gives what level of proficiency is expected of the user, some general constraints while making the software and some assumptions and dependencies that are assumed. Section 3 gives specific requirements which the software is expected to deliver. Functional requirements are given by various use cases. Some performance requirements and design constraints are also given.

**2. Overall Description:**

* 1. **Product Perspective:**

1. The web pages are present to provide the user interface on customer client side. Communication between customer and server is provided through URL.
2. It is a Window Based Platform. It works on any operating system.
3. The project is user-friendly.
4. This system will reduce the manual operation required to maintain all the records of Police department.
5. CRM is intended to be a stand-alone product and should not depend on the availability of other software.
   1. **Product Functions:**

|  |  |  |
| --- | --- | --- |
| Class of use cases | Use cases | Description of use cases |
| Use case related to  Installation | Installation | CRM installation |
| Use case related to  System Authorization | Login | 1. Login into the account 2. Change of password |
| Change password |
| Use case related to  Register | Registration | Registration of new criminals |
| Use case related to  Search | Search | Search the criminal is already present in the database or not |
| Use case related to  Report | Generating Reports | To record the already existing criminals |
| Use case related to  Delete | Delete | Delete the criminal record if the crime done by him doesn’t matter a lot. |
| Use case related to  Securities | Create security | Create a new security |
| Rename Security | Rename an existing security |
| Delete Security | Delete the existing security |
| Use case related to  Information display | View data | Show details of criminals |
| Use case related to release dairy | Viewing releasing dairy | Information of release date of prisoner |
| Use case related to confirm interview request | Confirm interview request | All interview requests by relatives of  Prisoners |

**2.3 User characteristics:**

a. The user should be familiar with the Criminal Record Management related terminology .

b. The user should know the basic details of criminals such as name/blood group.

c. Operator should be computer literate.

d. There will be a screen to display various information related to FIR.

e. A login screen for entering username, password will be provided.

f. There will be a screen for taking and modifying the information of the terms related to FIR.

**2.4 Principal Actors:**

The two principal actors of CRM are “user” and “system”.

**2.5 General Constraints:**

a. For full working CRMS may or may not require Internet connection.

b. CRMS is multi-user software.

c. It works on any Windows Operating system.

**2.6 Assumptions and Dependencies:**

1. Users are related to different branch of police Departments.

2. The information entered in Records should be correct.

3. User can fill the registration form.

**3 Specific Requirements:**

**3.1 Functional Requirements:**

We describe the functional requirements by giving various use cases.

**3.1.1 Functional requirement 1: Registration of users**

Introduction:

User has to register in order to view the criminal details, update records, searching the criminals, he may also post notifications.

Input:

The user has to fill the registration form. Registration form contains various text fields asking various details regarding the users like name, aadhar number etc.

Processing:

After filling all the fields in the registration form, user has to click on the submit button. When he clicks the submit button all the details provided by the user are saved into the database.

Output:

A dialog box will appear on the user screen, having the content “Registration Successful”.

**3.1.2 Functional requirement 2: Login**

Introduction:

Users at the time of registration, he will also provide unique user id and password. He will be logged into his account only by using this user id and password.

Input:

In the login form, user provides his user id along with his password. Then he clicks the login button.

Processing:

User has to provide unique user id and password at the time of registration in order to access his account. Those user id and password are saved in the database and whenever user clicks login button, administrator checks whether that user id and password are correct (or) incorrect. If they exist in the database, then he will be directed to his account.

Output:

If the user id and password are correct then user is redirected into his account, otherwise a alert message will be given to the user.

**3.1.3 Functional requirement 3: Registration of criminal**

Introduction:

Registration of criminal facilitates users to search the criminals and to know the details of criminals easily.

Input:

Criminals personal details like criminal no, age, crime type, occupation, address, most wanted.

Processing:

Validation checks are performed on the input data by the user/administrator like:

* Mandatory fields should not be kept blank.
* Prisoner family members should be check.
* All the entries in database should be correct.

After entering the details the user clicks on the submit button, then the information is stored in the database.

Output:

After clicking on the submit button, a dialog box containing “Registration Successful” will be displayed on the screen. Otherwise, it displays an error message.

**3.1.4 Functional requirement 4:**

Introduction:

Add or updating of records.

Input:

Username, password.

Administrator: Secret key.

Processing:

After entering the required information like punishment level, updating of records etc the various validation checks are performed. On submission of the information the updations are made to the database. Administrator must enter the correct key to update the records in the database.

Output:

The updated information is reflected in the stored database.

**3.1.5 Functional requirement 5:**

Introduction: Viewing data

Input:

User must be logged in. User name & password are required.

Processing:

On clicking the view button the system asks the user what he wants to view in the database. After entering, the system search for the specified database stored and displays the data on the screen.

Output:

If the specified database is stored then the whole database is displayed in the tabular form. The data displayed is updated.

**3.1.6 Functional requirement 6:**

Introduction: Record

Input:

Fir no, complaint no, Fir date, Fir type, information type, place of crime, act for crime, district of record, address of victim, passport no, name of police, received time, information received.

Processing time:

After inputting data, validation checks on various fields are performed. On submission of the information the record is searched in the stored database on the basis of the information submitted.

Output:

If the information submitted by the user is valid and found in the database then the corresponding record is displayed.

**3.1.7 Functional requirement 7:**

Introduction: Search

Input:

User must be logged in using user name & password.

Processing:

Search the criminal is already present in the database or not. After login, the user clicks on the search box. System asks the user for the details of the criminal. User enters name/blood group/jail no/DNA/image/sketch and clicks on the search button.

Output:

If the searched criminal is present in the database, information related to the criminal will be displayed. If not present in the database it displays an error message called “not found” on the screen.

**3.1.8 Functional requirement 8: Change password**

Introduction:

This is required for the users who lost their password then the user can change the password with the help of this page.

Input:

User name and password he remembers.

Processing:

User initiates the password change command. User is prompted for the old password, new password, and confirm new password. After filling all the fields he clicks on the submit button, then the new password will be stored in the database. User gets a message to his mail that password was changed.

**3.1.9 Functional requirement 9: Transfer**

Introduction:

This is required to transfer the criminal from one place to another. It is required to track the location of the criminal placed in jail.

Input:

The criminal name and the present location of the criminal and the location name to be transferred should be provided.

Processing:

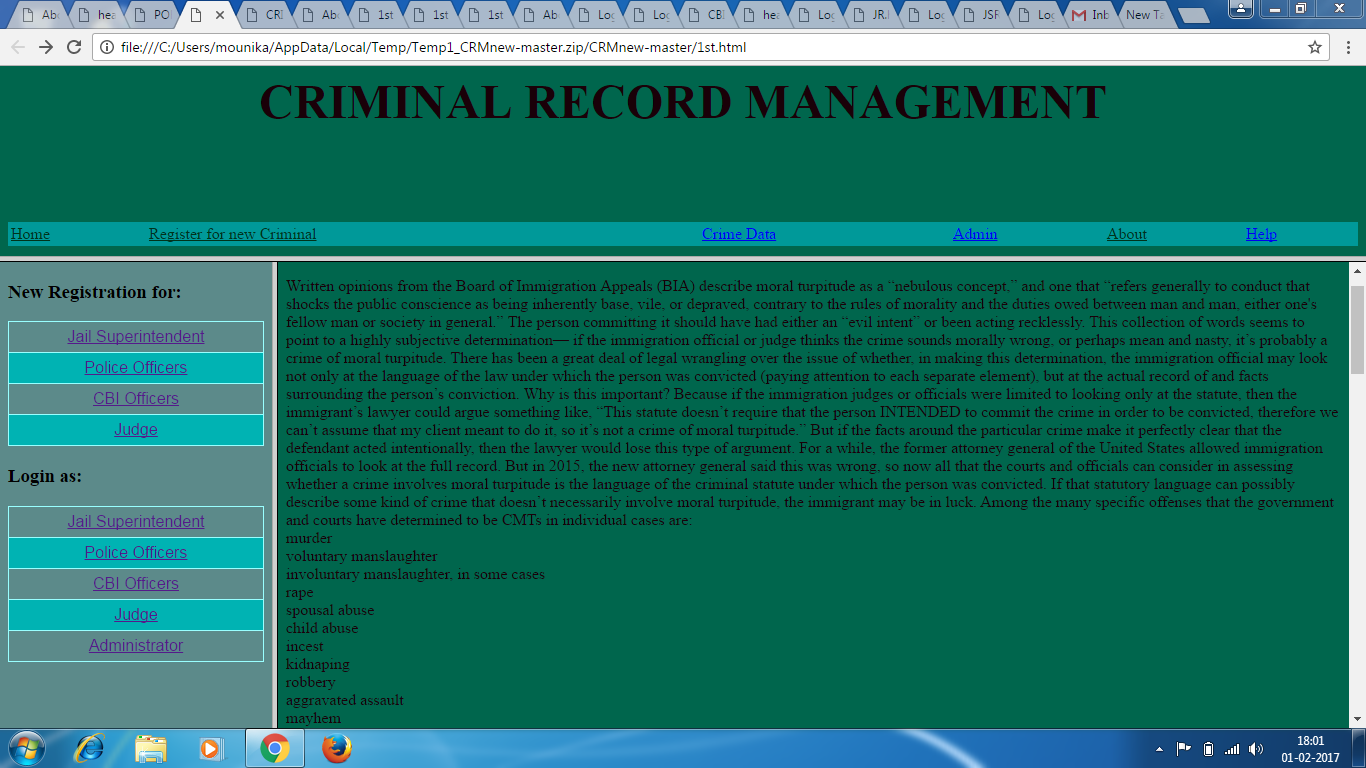
If the crime of the criminal is proved and it is very severe then the criminal is moved from one location to another location.

Output:

The details of location of the criminal will be updated and stored in the database.

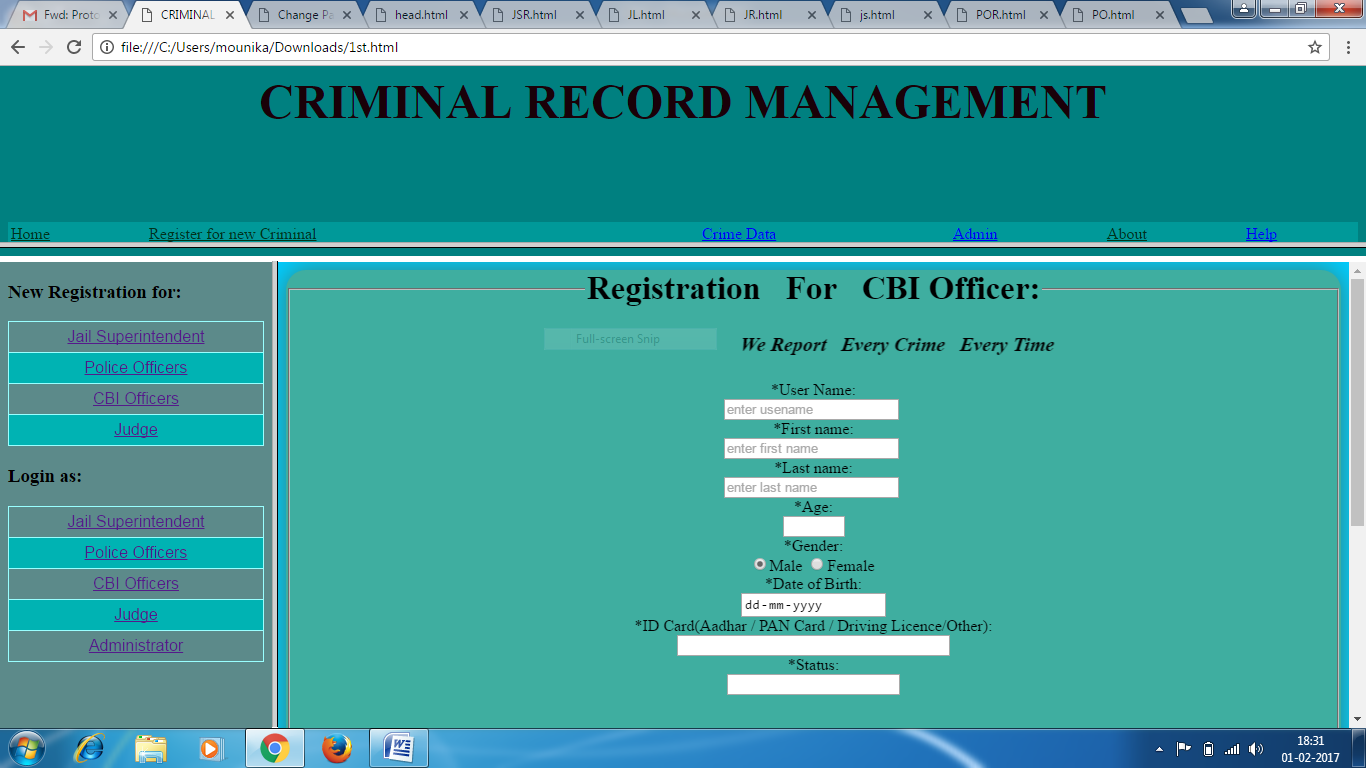
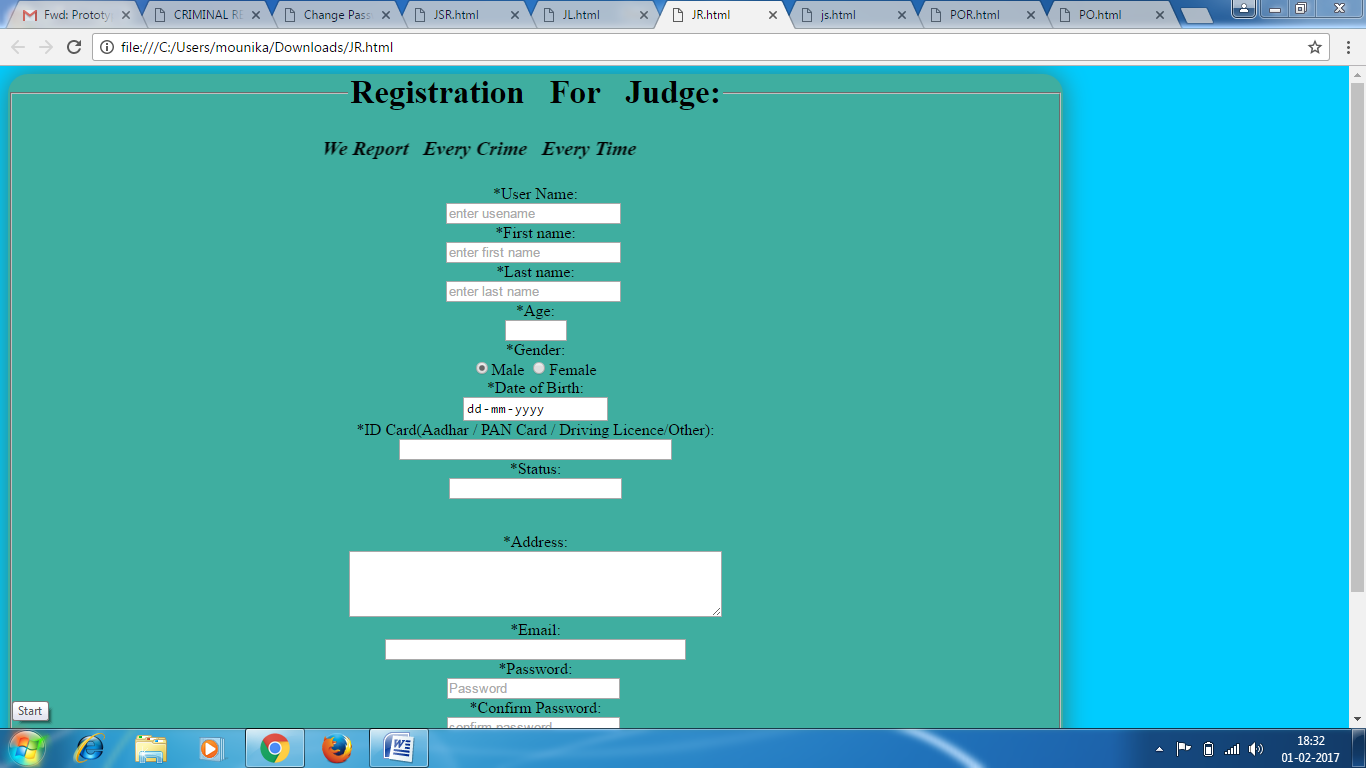
**3.2 External Interface Requirements:**

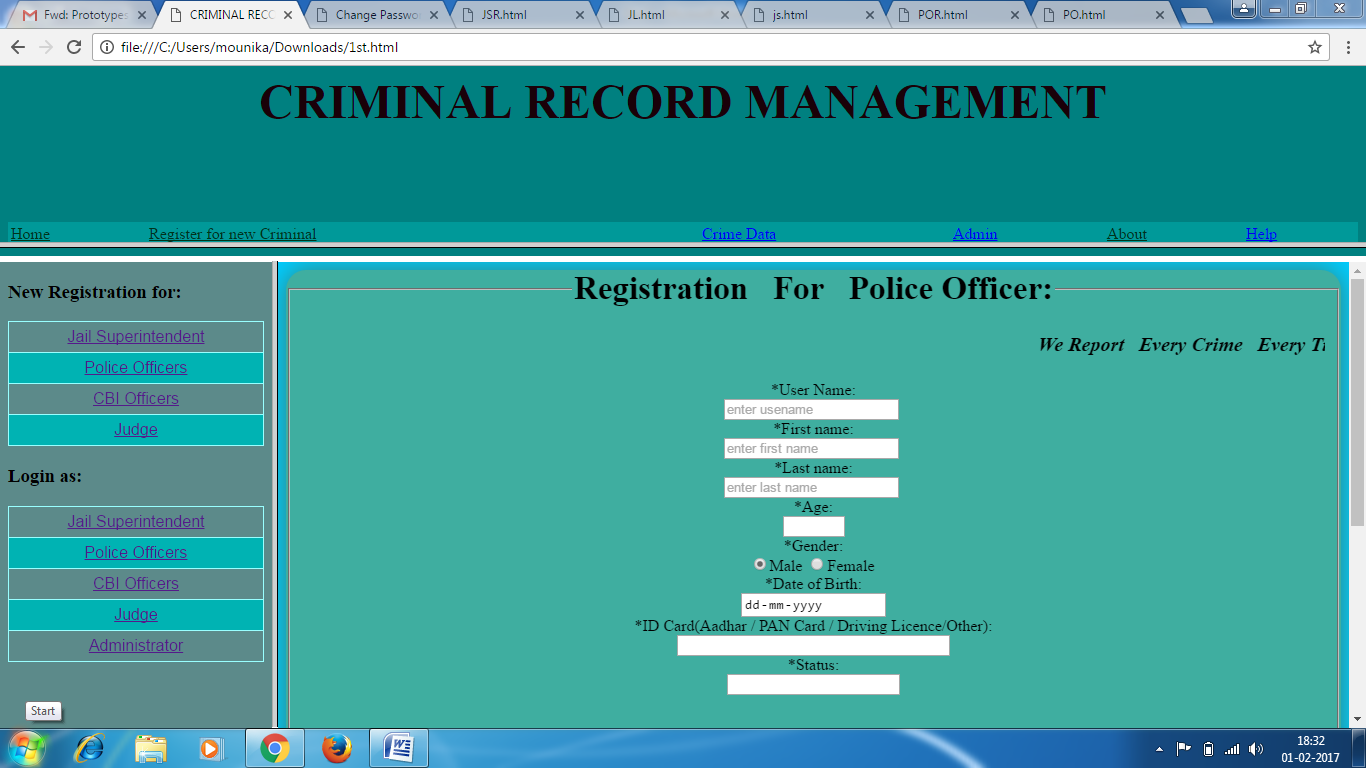
The external interface is a dynamically generated web page with professional graphics

****

The above prototype describes the homepage of criminal record management. The left frame consists of registrations and logins of the users. The right frame gives the description of the project. The top frame consists of some hyperlinks to navigate to another pages.

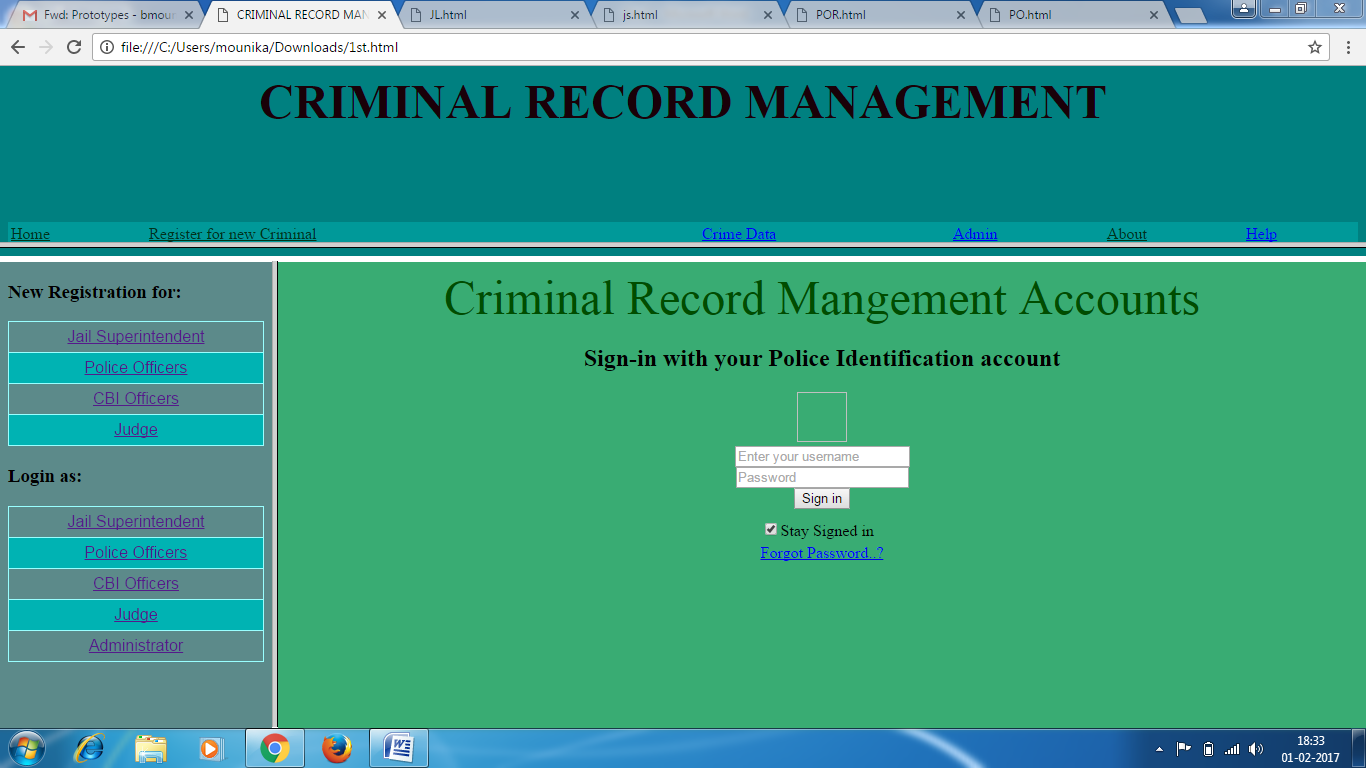


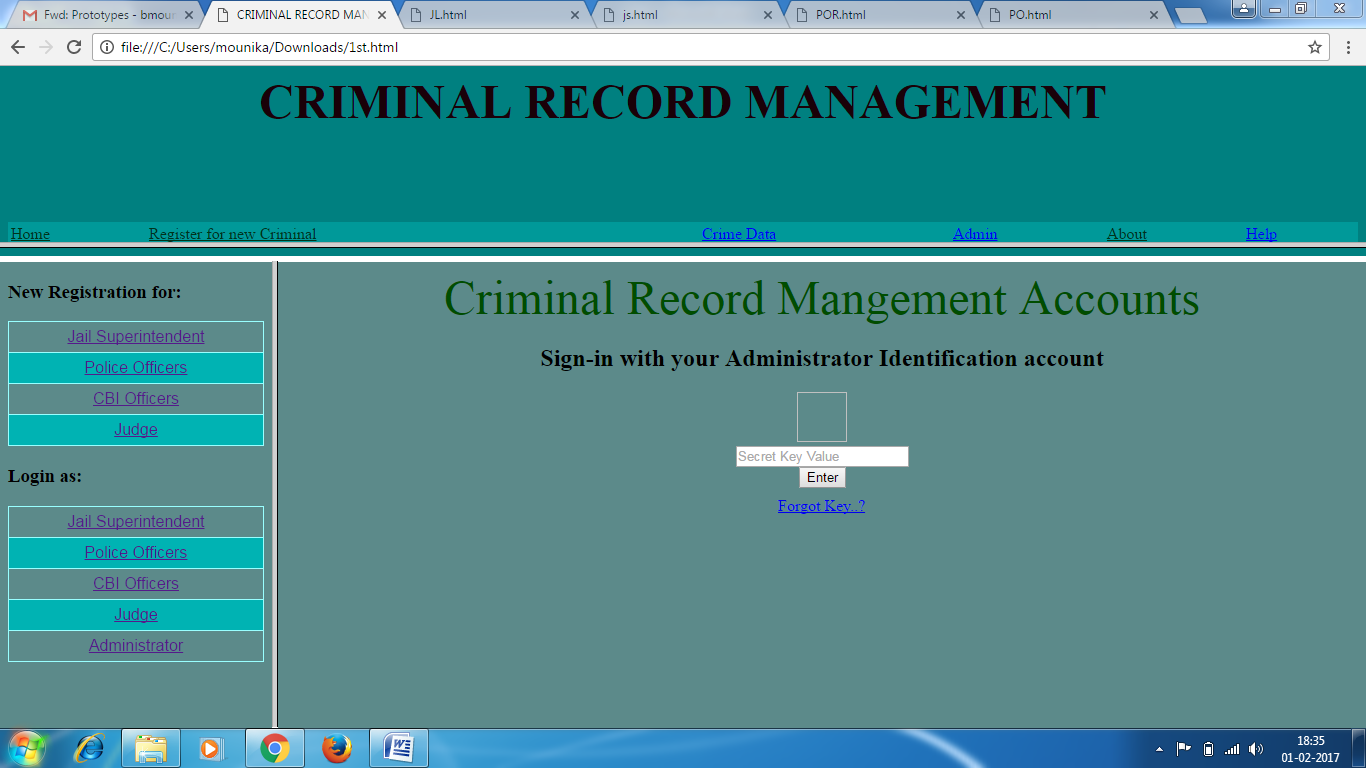
 



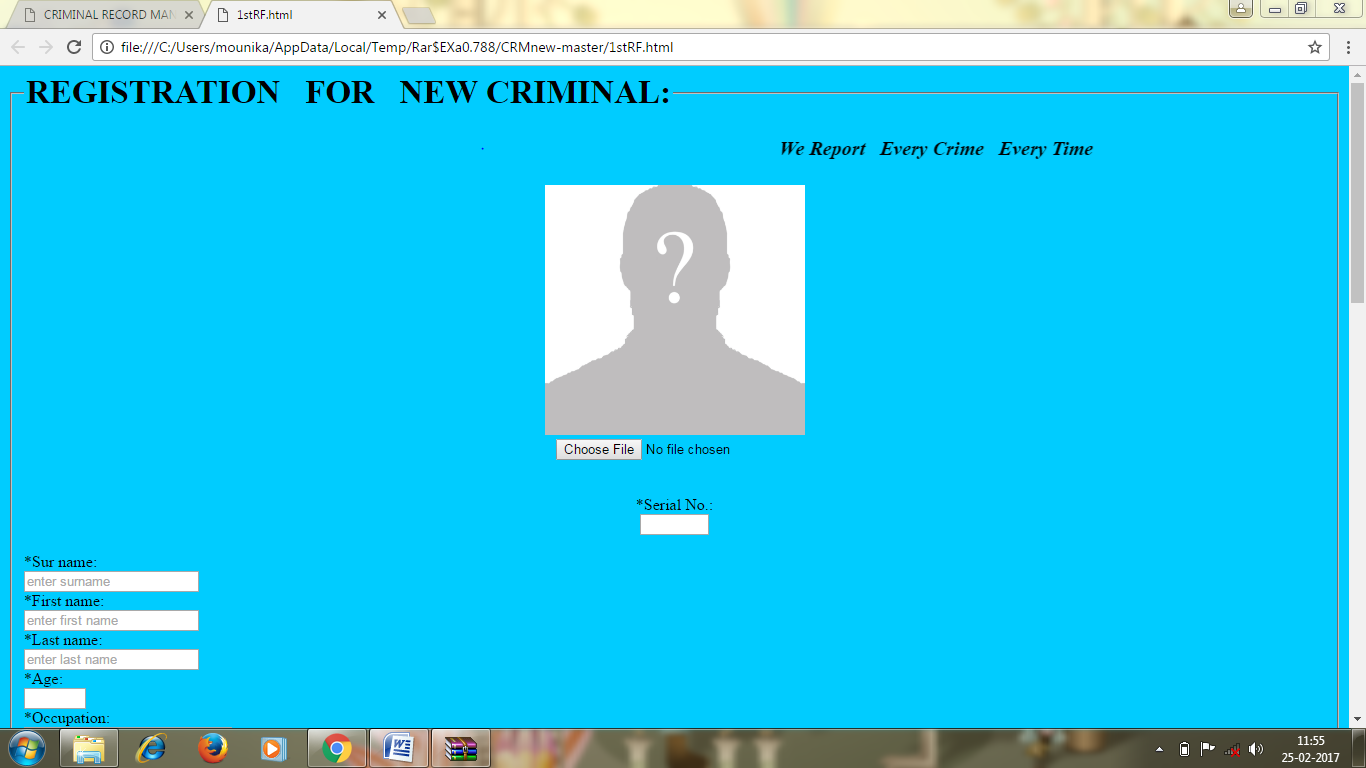
The above four screens shows the registration pages of users. The form takes the input fields like username, first name, last name, age, gender etc. All the fields should be filled mandatory. After filling all the fields the user clicks on the submit button. Then the user is displayed with message “Registration Successful”. Then the data will be stored in the database.

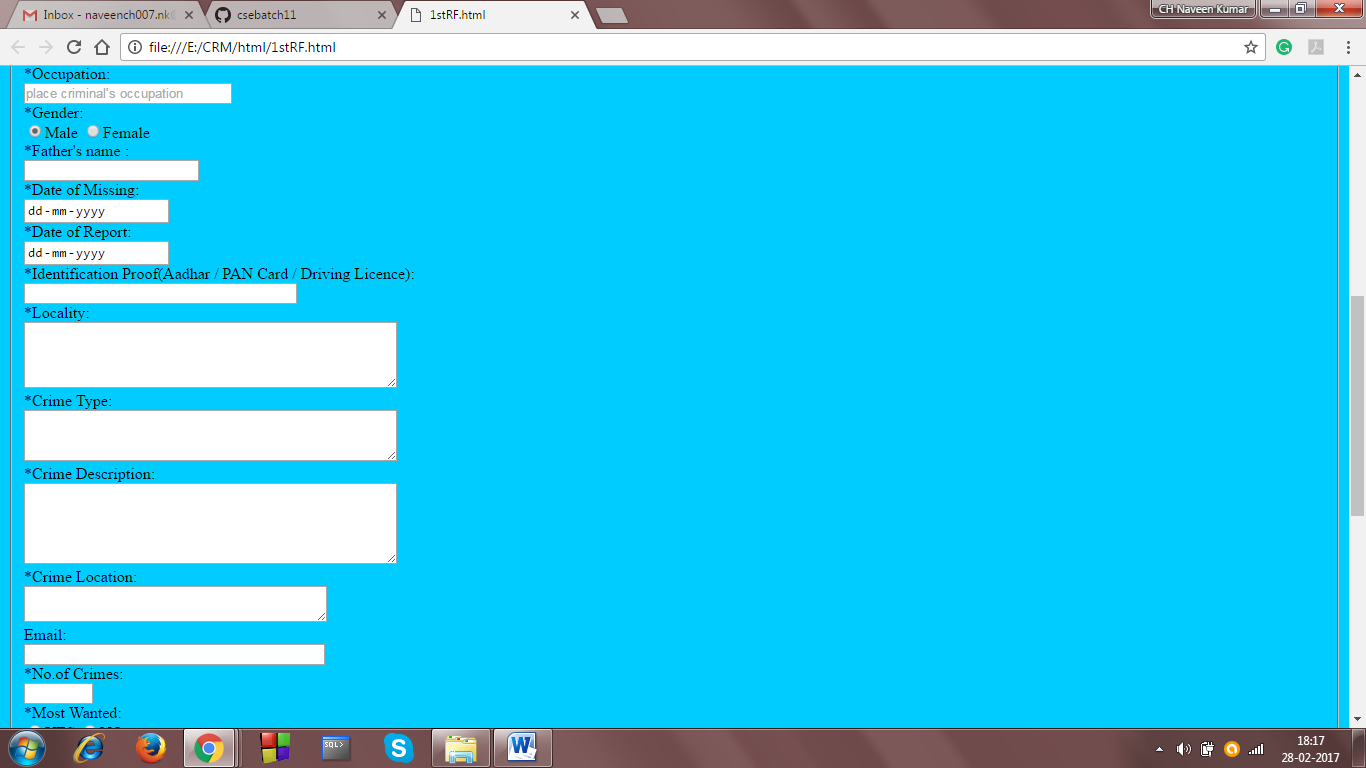




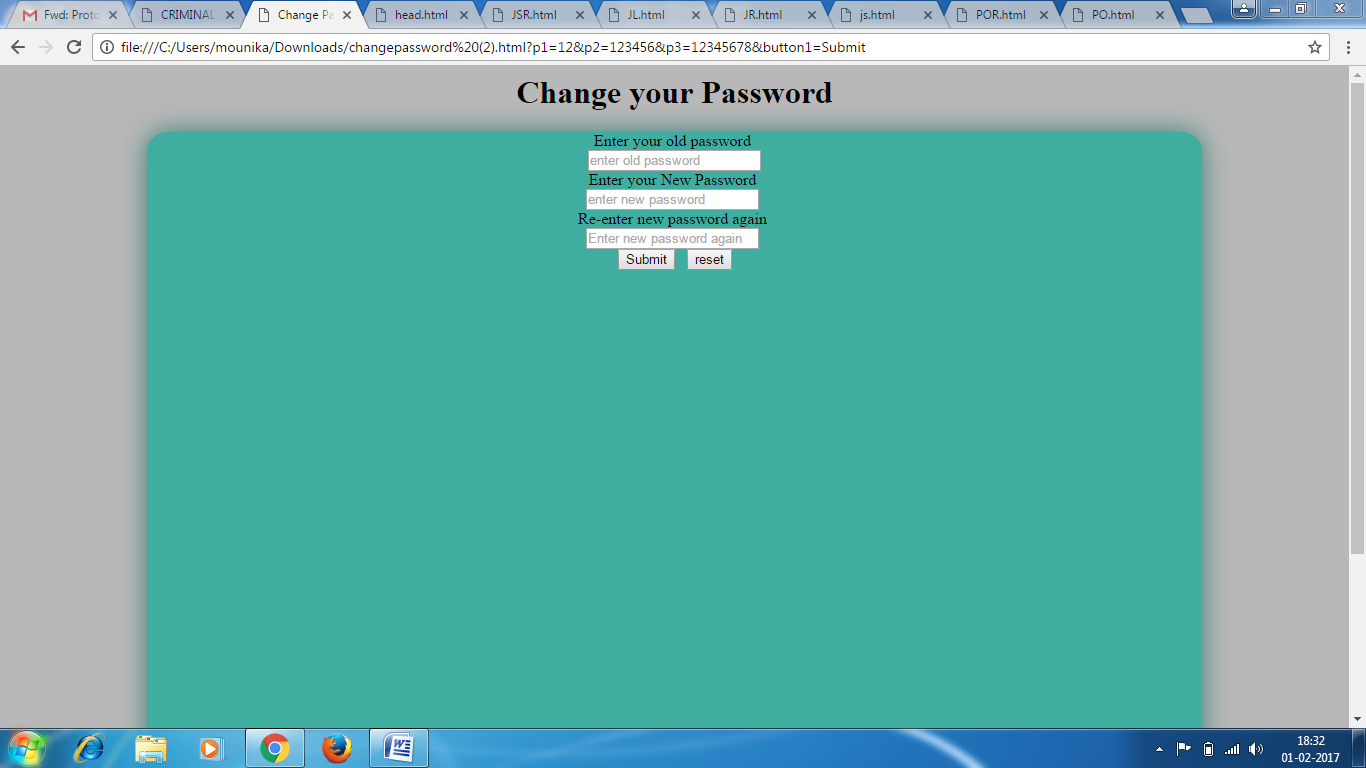
The above screen shows the login pages of users. The form takes the input fields like user name and password and secret key for administrator. After filling all the fields the user clicks on the submit button. If they are correct, he will be allowed to the next page. They should be matched with the data stored in the database.







The above screen shows the registration page of criminal. The jail superintendent has to fill all the fields in the form. The form takes the input fields like name, date of missing, Id proof, time and place of crime etc. After filling all the fields the user clicks on the submit button. Then the data will be stored in the database.



The above screen allows the user to change the password if the user has forgotten his/her password or if the user feels that it is unsecured. There are three text boxes names as old password, new password, confirm new password. The user is initially prompted to give the old password and then new password and he again asked to confirm new password. If new password and confirm new password are same and if user clicks on submit button, then new password will be updated in the database.

* 1. **Performance Requirements:**

1. Should run on 500 MHz, 256 MB machine.
2. 90% of the responses should be within 2 sec.
3. Responses to view information shall take no longer than 5 seconds to appear on the screen.
4. The system should be easy to handle.

**3.4 Design Constraints:**

1. Security: The files in which the information regarding securities and portfolios should be secured against malicious deformations.

2. Fault Tolerance: Data should not become corrupted in case of system crash or power failure.

* 1. **Security Requirements:**

1. System will use secured database to maintain records.
2. System will have different types of users and every user has access constraints.
3. Data should be secured and it should not be leaked outside.

**3.6 Maintainability Requirements:**

1. Maintenance of Records up-to-date.

2. Database backup and DDL Script.

3. Our application should be designed to be easily be maintained and repaired.

4. Changes must be verified once per day atleast.

**3.7 Reliability Requirements:**

Reliability is the probability that the system will be able to process all work correctly and completely without being aborted. Reliability is evaluated as follows:

1. System shall be recovered within 10 minutes if it is down.
2. System shall be recovered without intervention at user terminal if it is down.
3. System shall have 99% reliability during users operating hours.

**3.8 Availability Requirements:**

1. Online or Offline help to above said users, Application deployment executive and developer.

2. 24\*7 availability.

3. Secure access of confidential data. SSL can be used.

**3.9 Database Requirements:**

1. Use of GUI like JASPER to show strategic data to admin.

2. Database should only accept consistent data.

3. Updations must be done properly.

4. Crashing of data must be avoided.

**3.10 Documentation Requirements:**

1. Simplicity of interface.

2. Printouts for employment disbursal and history etc.

**3.11 Safety Requirements:**

1. System use shall not cause any harm to human users.

2. The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

**3.12 Operational Requirements:**

Once the implementation plan is decided, it is essential that user of the system is made familiar and comfortable with the environment. Users have to be made aware that what can be achieved with the new system. The user of the system should be given general idea of system before he uses the system.

**STRUCTURED ANALYSIS FOR CRIMINAL RECORD MANAGEMENT**

**DATA FLOW DIAGRAM**

**DATA ORIENTED ANALYSIS FOR CRIMINAL RECORD MANAGEMENT**

**ER-DIAGRAM**

****

**OBJECT ORIENTED ANALYSIS FOR CRIMINAL RECORD MANAGEMENT**

**USE CASE DIAGRAM**

****

**USE CASE SCENARIOS:**

**Use case related to installation:**

**Use Case 1**: Installation

**Primary Actor**: Administrator

**Pre-condition**: Internet connection available.

**Main Scenario**:

* + - 1. User initiates CRM installation program.
      2. System asks the user for the home directory in which all the working Files will be created.
      3. User is also asked for the initial login and password.
      4. User specifies the home directory and login/password.
      5. System creates the working files in the specified home directory.

Working files contain:

a. Information of Criminal.

b. Current Location of Jail.

c. Backup & Restore of data.

**Alternate Scenario**:

a. Network failure.

b. Installation aborted.

**Use cases related to system authorization:**

**Use Case 2:** Login

**Primary Actor**: Jail Superintendent, Police Officer, CBI Officer, Administrator, Judge

**Pre-condition**: Internet connection available

**Main Scenario**:

1. Start the application. User prompted for login and password.

2. User gives the login and password.

3. System does authentication.

4. Main screen is displayed.

**Alternate Scenario**:

1. Authorization fails.

2. Prompt the user that he typed the wrong password.

3. Allow him to re-enter the password. Give him 3 chances.

**Use case 3: Change Password**

**Primary Actor**: Jail Superintendent, Police Officer, CBI Officer, Administrator, Judge

**Pre-condition**: Certain User logged in

**Main Scenario**:

1. User initiates the password change command.

2. User is prompted for old password, new password and confirm new

Password.

3. User gives the old password, new password and confirm new

Password.

4. System does authentication.

5. New password is registered with the system.

6. User gets a message to his mail-id and mobile that password was changed.

**Alternate Scenario**:

1. Authorization fails.

2. Prompt the user that he typed the wrong password.

3. Allow him to re-enter the password. Give him 3 chances.

4. New password and confirm new password do not match. 4.1 Allow him to re-enter the attributes. Give 3 chances.

**Use case related to Registration:**

**Use case 4:** Registration

**Primary Actor**: Jail superintendant

**Pre-condition**: Internet connection available.

**Main Scenario**:

* + - 1. The visitor accesses the registration page for new ID.
      2. He/She fills up the criminal details like name/blood group/type of crime etc. and submits.
      3. The completeness of data is checked on client side.
      4. The Database is updated.

**Alternate Scenario**:

* + - 1. The data completeness check fails and the user is prompted to

Provide all details.

* + - 1. The database update fails.

**Use case related to search:**

**Use case 5:** Search

**Primary Actor**: Jail Superintendent, Police Officer, CBI Officer, Judge

**Pre-condition**: Certain User logged in.

**Main Scenario**:

1. User clicks on search box

2. System asks the user for the name of Criminal.

3. User enters name/blood group/Jail no/DNA/Image.

4. Click on the search button

**Alternate Scenario**:

1. Criminal name does not exist.

2. Criminal name exists but finger print /blood group does not match.

**Use case 6: Record**

**Primary Actor**: Administrator

**Pre-condition**: Administrator logged in

**Main Scenario**:

1. Maintain the database

2. Grant role to other user

3. Revoke role from other users.

4. Back and restore the data

**Alternate Scenario**:

1. Data missing of Certain Criminal

2. Backup and restore is done every day

**Use case related to delete:**

**Use case 7:** Delete data.

**Primary Actor**: Administrator

**Pre-condition**: User logged in.

**Main Scenario**:

1. User initiates the "delete data" functionality.

2. System asks for the name of the data.

3. The data is deleted.

**Alternate Scenario**:

1. Data does not exist.

2. Deletion fails, error message is displayed.

**Use cases related to securities:**

**Use case 8:** Create a security.

**Primary Actor**: Administrator.

**Pre-condition**: User logged in.

**Main Scenario**:

1. User selects the data in which the security is to be created.

2. User initiates the "create security" functionality.

3. System asks the user to enter the attributes of the security.

4. User specifies the following fields:

a. Name of Criminal or place

b. Type: where the Criminal is

5. An empty security of specified attributes is created.

**Alternate Scenario**:

1. A security with the given name already exists.

2. Security creation fails, error message is displayed.

**Use case 9: Rename security.**

**Primary Actor**: Administrator

**Pre-condition**: User logged in.

**Main Scenario**:

1. User selects the portfolio.

2. User initiates the "rename security" functionality.

3. System asks the user for the security and the new name.

4. User enters the security name and the new name.

5. Name is changed.

**Alternate Scenario**:

* + - 1. The security whose name is supposed to change does not exist.
      2. Renaming fails, the error message is displayed.
      3. Security with the same new name exists.
      4. Renaming fails, the error message is displayed.

**Use case 10: Delete security.**

**Primary Actor**: Administrator

**Pre-condition**: User logged in.

**Main Scenario**:

1. User selects the portfolio.

2. User initiates the "delete security" functionality.

3. System asks for the security name.

4. Security is deleted.

**Alternate Scenario**:

1. Security does not exist.

2. Deletion fails, error message is displayed.

**Use cases related to information display:**

**Use case 11: Display Information**

**Primary Actor**: User.

**Pre-condition**: User logged in.

**Main Scenario**:

* + 1. User clicks on search box.
    2. System asks the user for the name of Criminal.
    3. User enters name/blood group/Jail no/DNA/Image.
    4. Click on the display button.
    5. Then the details of the criminal are displayed.

**Alternate Scenario**:

1. Criminal name does not exists.

2. Criminal name exists but finger print /blood group does not match.

**Use case 12: Display Security**

**Primary Actor**: User

**Pre-condition**: User logged in.

**Main Scenario**:

1. User selects the option of viewing a particular security.

2. System displays all the transactions.

**Use case related to view release dairy:**

**Use case 13:** View release dairy

**Primary Actor**: Administrator, Police Officer

**Pre-condition**:

1. User must be logged in

2. He/she has to be at his home page

**Main Scenario**:

1. Retrieved the release diary information from the data base.

2. Viewing of data.

**Alternate Scenario**:

1. Retrieval of data failed.

**Use case related to confirm Interview confirmations:**

**Use case 14:** Confirm Interview request

**Primary Actor**: Administrator

**Pre-condition**:

Administrator should be logged in to his account to access this option

**Main Scenario**:

1. Verification status is checked

2.If OK then it is approved.

3.The database is updated

**Alternate Scenario**:

The interview request is not approved.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test scenario | Test case | Pre condition | Test steps | Test data | Expected result | Actual result | Post condition |
| Searching the criminal | Check response on entering the valid name and proof | The data of the criminal should be available | 1.Launch the application  2.enter the criminal name and proof | Criminal name=”john”  Aadhar:43630304052 | “details of the criminal must be printed” | “details of criminal is printed” | “pass” |
| Searching the criminal | Check response on entering the invalid name and proof | The data of the criminal should be available | 1.Launch the application  2.enter the criminal name and proof | Criminal name=”joh”  Aadhar:4363030452 | “enter a valid name and aadhar number” | “enter a valid name and aadhar number” | “error” |
| Searching the criminal | Check response on entering the criminal name | The data of the criminal should be available | 1.Launch the application  2.Fill all the details  3.click on the submit button | Criminal name=”john” | “enter the id proof” | “id proof also must” | “error” |
| Searching the criminal | Check response by entering the criminal name whose punishment level is completed and left the jail | The data of the criminal should be available | 1.Launch the application  2. Fill all the details | Criminal name=”john”  Aadhar:43630304052 | “details of criminal must be correct” | “details of criminal not found in the database” | “error” |