**Android-Based Staff and Offender Records Management System for Police Department**

**Software Requirements Specification**

**Group Id: Bc200401612**

**Supervisor Name: Irshad Nasir**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date (dd/mm/yyyy)** | **Version** | **Description** | **Author** |
| 2 Dec 2023 | 1.0 | The Android-Based Staff and Offender Records Management System for Police Department is an Android application designed to modernize police records. It enables secure user authentication, data input at police stations, hierarchical access to records, and report generation. Administrators can manage users, locations, and data, making it a comprehensive tool for efficient and secure police record management. The system is developed using Android Java/Kotlin, XML, and integrates with online databases like Firebase or MySQL. | Bc200401612 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. Scope (of the project)
2. Functional Requirements Non-Functional requirements
3. Use Case Diagram
4. Usage Scenarios
5. Adopted Methodology
6. Work Plan (Use MS Project to create Schedule/Work Plan)

**SRS Document**

**Scope of Project:**

The scope of the project encompasses the development of an Android application to facilitate the efficient management of staff and offender records within the Police Department. The system aims to digitize and centralize the information retrieval, data input, and reporting processes, providing a comprehensive solution for police personnel at various levels of management.

**1.1 In-Scope Features**

**1.1.1 User Authentication**

The application will include a secure authentication mechanism to ensure that only authorized users, including police officers and administrators, can access the system.

**1.1.2 Information Retrieval**

Users will be able to search for detailed information regarding Districts, Tehsils, and individual Police stations. This information will include data about staff and criminals associated with each location.

**1.1.3 Data Input and Management**

Police stations will have the capability to input and manage staff and criminal data. This includes updating existing information, adding new records, and deleting outdated or erroneous entries.

**1.1.4 Reporting**

Users, at different levels of management, will be able to generate detailed reports. Reports can be generated at the police station level, Tehsil level, and District level, providing insights into staff and criminal statistics for informed decision-making.

**1.1.5 Administrator Functionality**

Administrators will have exclusive access to functionalities such as user management, where they can assign, update, or delete the usernames and passwords of end-users. They will also have control over District, Tehsil, and Police station management, allowing them to add, edit, or delete entries.

**1.2 Out-of-Scope Features**

**1.2.1 Offline Functionality**

The system will primarily operate online to ensure real-time data synchronization and accessibility. Offline functionality is considered out of scope for this project.

**1.2.2 Mobile Device Compatibility**

While the application will be developed for Android devices, compatibility with other mobile platforms such as iOS will be considered out of scope.

**1.2.3 Advanced Analytics**

In-depth analytical features beyond basic reporting, such as predictive analytics, are considered out of scope for the initial version of the application.

**1.3 Constraints**

**1.3.1 Network Connectivity**

The application's functionality relies on a stable network connection. Users must have access to the internet or a secure VPN to connect to the central system.

**1.3.2 Security**

While the application will incorporate authentication mechanisms, it will not be responsible for the security of the network infrastructure. Security measures at the network level are considered external to the scope of the project.

**1.4 Assumptions and Dependencies**

**1.4.1 Data Accuracy**

The accuracy of staff and criminal data depends on the information provided by individual police stations. The system assumes that data input is accurate and regularly updated.

**1.4.2 Database Management**

The project assumes the availability and proper functioning of the selected online database management system (Firebase, MySQL, etc.) for storing and retrieving data.

Conclusion

The Android-Based Staff and Offender Records Management System for Police Department, with its defined scope, aims to provide a robust and user-friendly solution for digitizing and centralizing police records. This project will contribute to improving the efficiency and decision-making capabilities of the Police Department by providing timely and accurate information at various organizational levels.

**2.Functioals and non-functionals requirements:**

**I. Functional Requirements**

**I.1 User Module**

**I.1.1 Authentication**

The system shall provide a secure login mechanism for users, requiring a valid username and password.

**I.1.2 Information Retrieval**

Users shall be able to search for detailed information regarding Districts, Tehsils, and individual Police stations.

Users shall have access to staff and criminal data at Tehsil and District levels.

**I.1.3 Data Input and Management**

Police stations shall have the capability to input and manage staff and criminal data.

Users shall be able to search, edit, or delete data at the police station level.

**I.1.4 Reporting**

Users shall be able to generate reports at the police station level.

Users shall be able to create reports at the Tehsil level.

Users shall be able to produce reports at the district level.

**I.2 Administrator Module**

**I.2.1 Authentication**

The system shall provide a secure login mechanism for administrators, requiring a valid username and password.

**I.2.2 User Management**

Administrators shall be able to assign/update/delete the username and password of end-users.

**I.2.3 District, Tehsil, and Police Station Management**

Administrators shall be able to add, edit, or delete the names of Districts, Tehsils, and Police stations.

**I.2.4 Data Management**

Administrators shall be able to view/edit/delete District, Tehsil-wise staff, and criminal data.

Administrators shall be able to view/edit/delete police station-wise staff and criminal data.

**I.2.5 Reporting**

Administrators shall be able to generate various reports about staff and criminals at different levels, including Police Station, Tehsil, or District level.

**II. Non-Functional Requirements**

**II.1 Performance**

The system shall respond to user inputs within 3 seconds to ensure a responsive user experience.

Reports generation time shall not exceed 10 seconds for any level of the hierarchy.

**II.2 Security**

User authentication shall be conducted securely to prevent unauthorized access.

Data transmission between the application and the database shall be encrypted using industry-standard protocols.

Access to sensitive information shall be role-based, ensuring administrators have exclusive rights.

**II.3 Reliability**

The system shall have a 99.9% uptime, ensuring continuous availability for users.

Data integrity shall be maintained, and the system should have backup and recovery mechanisms in place.

**II.4 Usability**

The user interface shall be intuitive, ensuring users can perform tasks with minimal training.

The application shall be compatible with various screen sizes and resolutions for a seamless user experience.

**II.5 Scalability**

The system shall be designed to handle an increasing amount of data and users without compromising performance.

The architecture should allow for easy integration with additional police stations and districts.

**II.6 Database Management**

The selected online DBMS (Firebase, MySQL, etc.) shall provide scalability, reliability, and efficient data storage and retrieval.

**II.7 Compatibility**

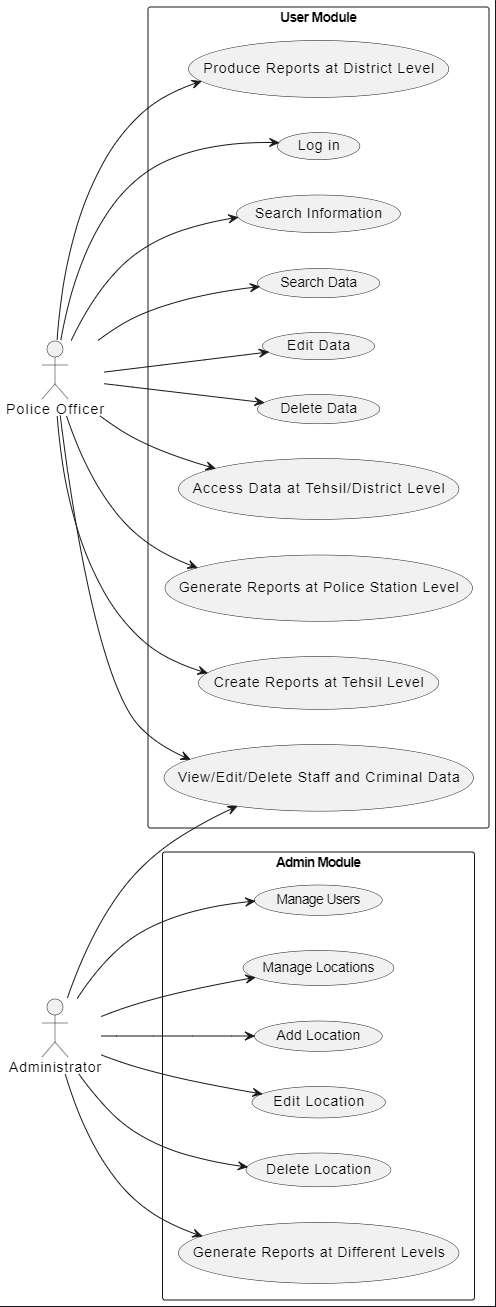
The application shall be compatible with Android devices running versions 6.0 (Marshmallow) and above.

The system shall be designed to accommodate future updates and improvements.

**II.8 Network Connectivity**

The system shall operate over standard internet connections or through secure VPNs for remote police stations.

**3.Use-Case Diagram:**

****

**4. Usage Scenarios:**

**User Model:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Title** | **Use Case ID** | **Actions** | **Description** | **Alternative Paths** | **Pre-Conditions** | **Post-Conditions** | **Author** | **Exceptions** |
| Login | UC001 | 1. Enter username and password. 2. Click "Login." | Authenticate user credentials and grant access to the system. | Invalid credentials prompt re-entry. | - User exists in the system. - System is operational. | User gains access to the system. | System Analyst | Invalid credentials trigger an error message. |
| Search Information | UC002 | 1. Select search criteria (District, Tehsil, Police Station). 2. Enter relevant details. 3. Click "Search." | Retrieve information about Districts, Tehsils, or Police Stations. | No results found. | - User is logged in.  - Data is available for the selected criteria. | Relevant information displayed. | System Analyst | No data found prompts a message. |
| Input/Manage Data | UC003 | 1. Select data type (Staff or Criminal). 2. Input or edit information. 3. Save changes. | Input or modify staff and criminal data at the police station level. | Cancel changes. | - User is logged in. - Editing existing data or adding new data. | Changes saved successfully. | System Analyst | Changes not saved prompt an error message. |
| Generate Reports | UC004 | 1. Select report type (Police Station, Tehsil, District). 2. Choose parameters. 3. Click "Generate." | Create detailed reports based on selected criteria. | No data for the selected parameters. | - User is logged in. - Relevant data is available. | Report generated successfully. | System Analyst | No data found triggers an informative message. |

**Admin Model:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Title** | **Use Case ID** | **Actions** | **Description** | **Alternative Paths** | **Pre-Conditions** | **Post-Conditions** | **Author** | **Exceptions** |
| Login | UC001 | 1. Enter administrator username and password. 2. Click "Login." | Authenticate administrator credentials and grant access to administrative functions. | Invalid credentials prompt re-entry. | - Administrator exists in the system. - System is operational | Administrator gains access to administrative functions. | System Analyst | Invalid credentials trigger an error message. |
| Manage Users | UC002 | 1. Select user management. 2. Choose action (add/update/delete). 3. Input details. 4. Save changes | Add, update, or delete user accounts. | Cancel changes**.** | - Administrator is logged in. - User management action selected | Changes to user accounts saved successfully. | System Analyst | Changes not saved prompt an error message. |
| Manage Districts/Tehsils/Police Stations | UC003 | 1. Select management of districts, Tehsils, or police stations.  2. Choose action (add/update/delete).  3. Input details.  4. Save changes. | Add, update, or delete entries for Districts, Tehsils, or Police Stations. | Cancel changes. | - Administrator is logged in. - Management action selected. | Changes to locations saved successfully. | System Analyst | Changes not saved prompt an error message. |
| Manage Data | UC004 | 1. Select data management. 2. Choose data type (District, Tehsil, Staff, Criminal).  3. Edit or delete data. 4. Save changes. | Edit or delete data for Districts, Tehsils, Staff, or Criminals. | Cancel changes. | - Administrator is logged in. - Data management action selected. | Changes to data saved successfully. | System Analyst | Changes not saved prompt an error message. |
| Generate Reports | UC005 | 1. Select report type (Police Station, Tehsil, District).  2. Choose parameters. 3. Click "Generate." | Create detailed reports based on selected criteria. | No data for the selected parameters. | - Administrator is logged in. - Relevant data is available. | Report generated successfully. | System Analyst | No data found triggers an informative message. |

**5.Adopted Methodology:**

**VU Process Model**

The VU (V-Model with an Upward Arrow) Process Model is a hybrid methodology that combines elements of both the waterfall and spiral models. It integrates the systematic and sequential nature of the waterfall model with the iterative and risk-driven aspects of the spiral model. The VU Process Model is particularly suitable for projects that require a well-defined structure like the waterfall model but also need the flexibility to accommodate changes and enhancements during the development process.

**1. Requirements Analysis and Specification:**

**1.1 Activities:**

Gather and analyze requirements from stakeholders.

Define detailed specifications based on the gathered requirements.

**1.2 Deliverables:**

Requirements document.

Detailed specifications document.

**1.3 Rationale:**

This phase sets the foundation for the entire project by clearly defining the functional and non-functional requirements.

**2. System Design:**

**2.1 Activities:**

Develop a comprehensive system architecture based on the requirements.

Create detailed design specifications for the application's components.

**2.2 Deliverables:**

System architecture document.

Detailed design specifications.

**2.3 Rationale:**

This phase ensures a clear understanding of the system's structure and design, facilitating a smooth transition to the implementation phase.

**3. Implementation:**

**3.1 Activities:**

Write code based on the detailed design specifications.

Conduct unit testing to ensure individual components function correctly.

**3.2 Deliverables:**

Executable code.

Unit test reports.

**3.3 Rationale:**

The implementation phase focuses on converting design specifications into functional code and ensuring the correctness of individual components.

**4. Integration and Testing:**

**4.1 Activities:**

Integrate individual components into a cohesive system.

Conduct system testing to ensure the integrated system meets specified requirements.

**4.2 Deliverables:**

Integrated system.

System test reports.

**4.3 Rationale:**

This phase verifies that the entire system operates as expected when all components are combined and ensures the overall system's quality.

**5. Acceptance Testing:**

**5.1 Activities:**

Collaborate with stakeholders to conduct acceptance testing.

Address and resolve any issues identified during testing.

**5.2 Deliverables:**

Accepted system.

Issues resolution documentation.

**5.3 Rationale:**

Acceptance testing involves verifying that the system meets stakeholder expectations and is ready for deployment.

**6. Maintenance and Enhancements:**

**6.1 Activities:**

Address post-deployment issues and bug fixes.

Incorporate enhancements or changes based on user feedback.

**6.2 Deliverables:**

Updated system version.

Maintenance documentation.

**6.3 Rationale:**

This phase supports ongoing improvements to the system based on user feedback and changing requirements.

**Iterative Aspects of the VU Model:**

The model allows for iterative development cycles within each phase, especially during implementation and testing, providing flexibility for adjustments and refinements.

Feedback loops are incorporated after each major phase, allowing for validation and adjustments before proceeding to the next stage.

**Rationale for Choosing the VU Process Model:**

The VU Process Model offers a structured approach similar to the waterfall model, ensuring clarity and control over project milestones.

The iterative nature of the model, inspired by the spiral model, accommodates changes and improvements throughout the development life cycle.

The upward arrow signifies the continuous improvement and enhancement of the system even after deployment, aligning with the dynamic nature of modern software development.

By adopting the VU Process Model, the project aims to balance the advantages of a well-structured development process with the flexibility needed to adapt to evolving requirements and feedback.

**6.Work Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tasks** | **Start Date** | **Duration** | **End Date** | **Dependencies** |
| Requirements Analysis | 4/12/2023 | 15 days | 19/12/2023 | ------------------- |
| System Design | 20/12/2023 | 12 days | 31/12/2023 | Requirements Analysis |
| Implementation | 1/1/2024 | 30 days | 30/01/2024 | System Design |
| Integration & Testing | 31/01/2024 | 20 days | 19/02/2024 | Implementation |
| Acceptance Testing | 20/02/2024 | 10 days | 29/02/2023 | Integration & Testing |
| Maintenance | 1/3/2024 | 20 days | 20/03/2023 | Acceptance Testing |