E-GOVERNANCE

MISSION MODE PROJECT (MMP)

CRIME & CRIMINAL TRACKING NETWORK AND SYSTEMS (CCTNS)

FUNCTIONAL REQUIREMENTS SPECIFICATION V1.0
(DRAFT)

सत्यमेव जयते

MINISTRY OF HOME AFFAIRS
GOVERNMENT OF INDIA

TABLE OF CONTENTS

1. INTRODUCTION

The Functional Requirements Specifications (FRS) report provides the detailed description of the functionalities required for the first version of the CCTNS. The key guiding principle behind the functional design of CCTNS V1.0 is to focus on the critical functionality that provides value to the police personnel at the cutting edge which in turn can improve the outcomes in the areas of "Investigation of Crime" and "Detection of Criminals".

2. FUNCTIONAL OVERVIEW

CCTNS V1.0 functionality is designed to focus on delivering value to IOs, records room staff and citizens within the broad crime investigation area. Based on the guiding principles stated above, nine different function blocks have been identified and the detailed functionality of each block was determined.
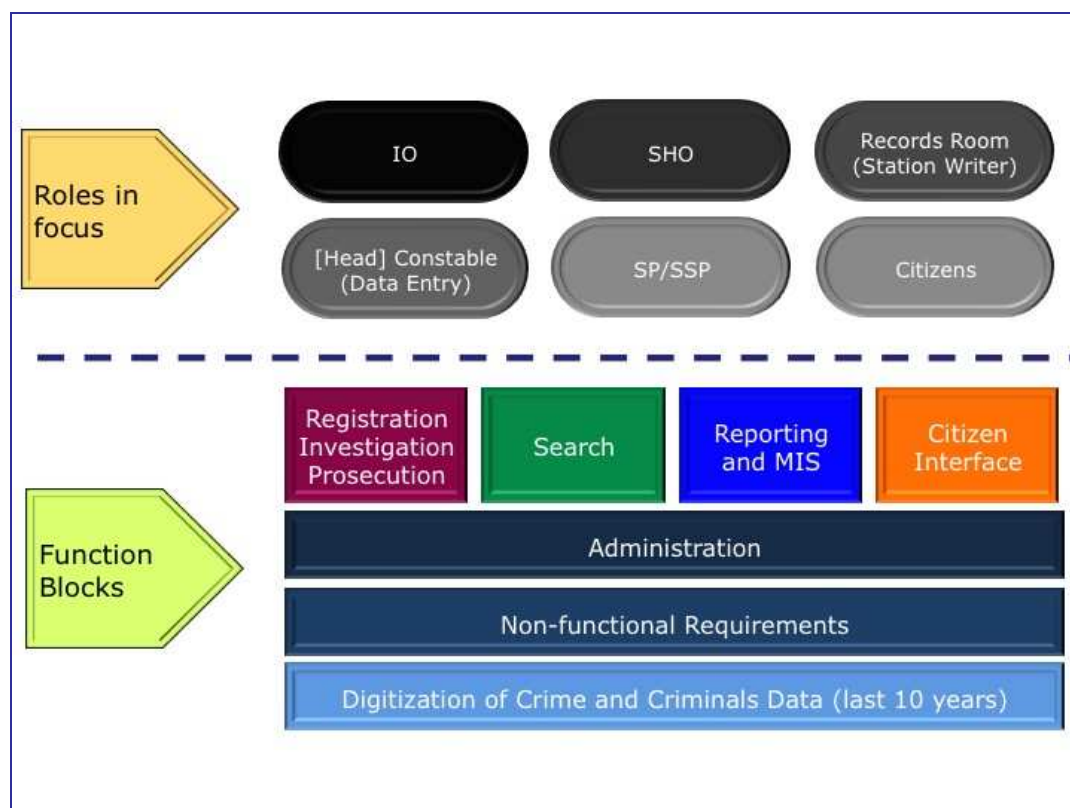
3. DESCRIPTION OF THE MODULES AND FUNCTIONAL REQUIREMENTS

The functionality of the CCTNS application is focused on providing value to the police personnel, especially the officers operating at the cutting edge and easing the day to day operations of the police function.

Registration

Citizens can register their complaints with police and then based on the evidence, facts and following investigation, police shall take the complaint forward. The Registration module acts as an interface between the police and citizens and it eases the approach, interaction and information exchange between police and complainants.

Investigation

After a complaint is initiated, police initiates the investigation process. The Investigation module of the CCTNS facilitates the investigation process and introduces operational efficiencies by automating most of the tasks that take place after initial entries are made during Registration.

Prosecution

Interfacing with the courts during the prosecution of cases is an integral part of the responsibilities of police personnel. A designated constable from each police station constantly interfaces with the courts. The Prosecution module of the CCTNS aids this interfacing by providing a platform to record entries of the court interactions.

Search

The Search module of the CCTNS gives police personnel the ability to execute a basic or advanced search on cases. Using the search functionality, police personnel can search for a particular person, type of crime, modus operandi, property etc. It also gives the user the ability to customize the results view by criminal/accused or by cases. It makes reporting easy for police by enabling them to execute different types of queries such as monthly reporting, RTI related etc.

Citizen Interface

The Citizen Interface module of the CCTNS acts as a conduit for the information exchange between citizens and police units/personnel. Citizens can use it as a tool to get information or acknowledgements from police. The police in turn can use it to respond to citizens with very little turnaround time. It improves overall productivity by helping citizens and police to cut short the drudgery of large amounts of paperwork.

Navigation

The Navigation module of the CCTNS provides role based landing pages which help in navigating through the CCTNS application. It shows information such as cases assigned, alerts, pending tasks etc hence helping police personnel to plan better and execute with greater efficiency.

Application Configuration

The Configuration module of the CCTNS helps keep the application configured according to the states' requirements in addition to keeping data elements/rules up to date. With a proper configuration, information such as act and sections, state specific data, castes, tribes, property information etc. can be created updated and deleted.

The functional requirements for each of the modules are provided as A1 to A7 in separate enclosures.

4. NON-FUNCTIONAL REQUIREMENTS

The non-functional requirements specify the qualitative attributes such as user-friendliness and performance of the system that are critical for the increased user-acceptance of the application.

Help Module	
1.	The solution should provide detailed context-sensitive help material for all the possible actions and scenarios on all user interfaces in the application.
2.	The help should be accessible to the users both in the offline and online mode

Support Module	
1.	The solution should provide an interface for the user to log any defects or enhancement requests on the application and track thereafter.
2.	The solution should send alerts (e.g., email, SMS) to the user if the user chooses to whenever any action has been taken on the alert
3.	The solution should enable the user to track the submitted defect or enhancement request.
4.	The solution should enable the help-desk user to view the reports on the submitted defects or enhancement requests category-wise, status-wise, and age-wise.
5.	The support solution should be accessible to the users both from within the application and also outside the application through a browser interface.

Audit Module	
1.	An audit trail is a record of actions taken by either the user or the system triggers. This includes actions taken by users or Administrators, or actions initiated automatically by the system as a result of system parameters. The System must keep an unalterable audit trail capable of automatically capturing and storing information about: • All the actions (create/read/update/delete) that are taken upon the critical entities (case, suspect, property,…) in the system • The user initiating and or carrying out the action; • The date and time of the event.

	• Administrative parameters
	The word "unalterable" is to mean that the audit trail data cannot be modified in any way or deleted by any user; it may be subject to re-department and copying to removable media if required, so long as its contents remain unchanged.
2.	Once the audit trail functionality has been activated, the System must track events without manual intervention, and store in the audit trail information about them.
3.	The System must maintain the audit trail for as long as required, which will be at least for the life of the case to which it refers.
4.	The System must ensure that audit trail data is available for inspection on request, so that a specific event can be identified and all related data made accessible, and that this can be achieved by authorised external personnel who have little or no familiarity with the system.
5.	The System must be able to export audit trails for specified cases (without affecting the audit trail stored by the System). This functionality can be used by external auditors who wish to examine or analyse system activity.
6.	The System must be able to capture and store violations (i.e. A user's attempts to access a case to which he is denied access), and (where violations can validly be attempted) attempted violations, of access control mechanisms.
7.	The System must at a minimum be able to provide reports for actions on cases organised: • By case; • By user; • In chronological sequence.
8.	The System should be able to provide reports for actions on cases organised by workstation and (where technically appropriate) by network address.

The requirements specify the requirements to control the user access to correspondences, files, and records and various functionalities provided within the system.

Access Module	
1.	The System must allow the user to limit access to cases to specified users or user groups.
2.	The system should provide for role-based control for the functionality within

		the system.
	3.	The System must allow a user to be a member of more than one group.
	4.	The System must allow only admin-users to set up user profiles and allocate users to groups.
	5.	The System should allow a user to stipulate which other users or groups can access cases.
	6.	The System must allow changes to security attributes for groups or users (such as access rights, security level, privileges, password allocation and management) to be made only by super-user.
	7.	If a user requests access to, or searches for, a case which he does not have the right to access, the System must provide one of the following responses (selectable at configuration time): • display title and metadata; • display the existence of a case but not its title or other metadata; • do not display any case information or indicate its existence in any way. These options are presented in order of increasing security. Note that the requirement in the third option (i.e. the most stringent) implies that the System must not include such cases in any count of search results; this level of security is normally appropriate for cases dealing with matters such as national security.
	8.	If a user performs a quick or advanced search, the System must never include in the search result list any record which the user does not have the right to access.
	9.	If the System allows users to make unauthorised attempts to access cases, it must log these in the audit trail.
	10.	Any access to cases, and all other activities involving the cases and related documents or data should also need to be stored in the audit trail to ensure legal admissibility and to assist in data recovery.

Ease of Use		
	1.	All error messages produced by the System must be meaningful, so that they can be appropriately acted upon by the users who are likely to see them. Ideally, each error message will be accompanied by explanatory text and an indication of the action(s) which the user can take in response to the error.
	2.	The System must employ a single set of user interface rules, or a small number of sets to provide a familiar and common look and feel for the application.

3.	The System must be able to display several entities (cases, suspects) simultaneously.
4.	The interfaces must be made customizable or user-configurable to the extent possible. (e.g., the displayed columns in the table, move, resize, modify the appearance). Such configurations must be saved in the user profile.
5.	The System user interface must be suitable for users with special needs; that is, compatible with specialist software that may be used and with appropriate interface guidelines
6.	The System must provide End User and Administrator functions which are easy to use and intuitive throughout.
7.	The System must allow persistent defaults for data entry where desirable. These defaults should include: • user-definable values; • values same as previous item; • values derived from context, e.g. date, file reference, user identifier;
8.	Frequently-executed System transactions must be designed so that they can be completed with a small number of interactions (e.g. mouse clicks).
9.	Where the System employs a graphical user interface, it must allow users to customise it. Customisation should include, but need not be limited to the following changes: • menu contents; • layout of screens; • use of function keys; • on-screen colours, fonts and font sizes;

Usability	
1.	The user interfaces should be designed to make them user-intuitive.
2.	The user interfaces of the system should comply with Standard ISO 9241.
3.	ICT accessibility: ISO 9241-20 shall be the standard for guidance on ICT accessibility. Application user interfaces to meet its requirements and recommendations. Software accessibility ISO 9241-171 shall be the standard for guidance on software accessibility. User interfaces should meet its requirements and recommendations. Content accessibility WCAG 1.0 shall be the standard used for guidance on content accessibility. The application logo to be available on all pages as a link to the home page.

4.	Providing text equivalents for non-text media objects: All non-text media objects, such as graphical images or video, should be provided with alternative equivalent textual descriptions and/or with equivalent text-based functionality.
5.	Making navigation self-descriptive: Navigation should be designed to help users understand where they are, where they have been and where they can go next. General guidance on achieving self-descriptiveness is given in ISO 9241-110.
6.	Showing users where they are: Each presentation segment (page or window) should provide the user with a clear and sufficient indication of where he or she is in the navigation structure and of the current segment position with respect to the overall structure.
7.	Offering alternative access paths: Alternative access paths for navigating to a specific unit of content should be offered to support different navigation strategies.
8.	Minimizing navigation effort: The number of navigation steps needed to reach a certain piece of content should be minimized as long as different mental models, navigation strategies and tasks of the user are taken into account.
9.	Splash screens should be avoided unless they provide useful content or feedback about the application state to the user. If a splash screen is used, a navigation option to skip it should be offered.
10.	Avoiding opening unnecessary windows: Additional windows such as new browser windows or pop-up windows should only be opened if this supports the user's task. Opening new windows can distract, confuse or impede users for a variety of reasons. They can superimpose the primary window, hiding relevant information. They could make it cognitively more difficult to understand the navigation structure with negative effects on both usability and accessibility. They also require additional user actions for closing unwanted windows.
11.	Vertical scrolling should be minimized. This may be done by placing important information at the top and providing links to information that is further down the page. Horizontal scrolling should be avoided wherever possible.
12.	Designing for input device independence: User interfaces should be designed to allow activation of controls by a variety of input devices. The ability to choose between different input devices for activating controls such as links, fields and buttons is important both for users who prefer a certain input mode, mobile users and users with disabilities. In general, device independence can be achieved if the functionality is operable via a keyboard.
13.	Making user interfaces robust: User interfaces should be designed to be as robust as possible in the face of changing technology. This encompasses being able to present content containing newer technologies by older user agents as well as

	designing content to be usable with future technologies.
14.	Acceptable opening / download times: Application pages should be designed and implemented so that there are acceptable opening times and download times for the expected range of technical contexts of use (e.g. bandwidth between the application and the user). This is particularly important for frequently accessed pages or pages that are important for user navigation and exploration, such as the home page.
15.	Minimizing user errors: Potential user errors as well as the effort needed to recover from errors should be minimized.
16.	Providing clear error messages: The content of error messages shown on the pages or special error pages should clearly state the reason why the error occurred and, if possible, actions the user can take to resolve the error. Users expect error messages to be in the same language as the user interface.
17.	Using appropriate formats, units of measurement or currency: When designing user interfaces for use by diverse groups, input and output of information elements such as currency, units of measurement, temperatures, date and time, phone numbers, address or postal codes should be designed so that they are usable.
18.	Making text resizable by the user: Text should be able to be resized by the user, using functions provided by the user agent or other appropriate means i.e. see ISO 9241-171.
19.	Text quality: The quality of textual content with respect to spelling and grammar should be sufficient so as not to impede readability.
20.	Writing style: The reading and understanding of the textual content on the screen should be supported by suitable means, including the use of short sentences, the division of the text into shorter chunks or the presentation of content items in the form of bullet points.
21.	Supporting text skimming: Fast skimming of text should be supported by the provision of clear links, bulleted lists, highlighted keywords, logical headings, and short phrases and sentences.
22.	Readability of text: Text presented on the pages should be readable taking into account the expected display characteristics and spatial arrangement. ISO 9241-303 shall be consulted for screen text legibility requirements.
23.	Distinguishable within-page links: Within-page links should be clearly distinguishable from other links that lead to a different page. EX. Within-page links are shown with dashed rather than solid underlines
24.	Avoiding link overload: Text pages containing large proportions of links should

	be formatted so that the presence of links does not impede the readability of the text.
25.	Using familiar terminology for navigation links: Navigation links — particularly links representing the main navigation structure — should be labelled with terms that are familiar to the user, based on his/her general knowledge, prior experience in the application domain or experience of using other systems.
26.	Using descriptive link labels: The target or purpose of a link should be directly indicated by its label, avoiding generic labels such as "go" or "click here" except where the purpose of the link is clear from its context on the page or the labels have commonly understood semantics in the particular application domain. Using appropriate terminology specific to the user's tasks and information needs is important for making the content easy to understand.
27.	Marking links opening new windows: Links that open new browser windows or pop-up windows should be clearly marked.
28.	Distinguishing navigation links from controls: Navigation links should be clearly distinguishable from controls activating some action. Typical action types in user interfaces include manipulating application data, performing searches, communication actions, such as opening a new e-mail window or starting a chat function, and Hpresentation-related actions, such as sorting a list of search results.
29.	Providing printable document versions: If a document is either too long, dispersed over several pages or in a specific layout that is not suitable for online reading, a printer-friendly version of the document should be provided that prints the content in a form acceptable to the user (e.g. in the expected layout, paper format, or orientation).
30.	Use of "white space": "White space" on a page i.e. space filled only with the background color should be used in such a way that it does not impair the visual skimming of the page. While white space is an important means of visually organizing the different content elements on a page, if the distance between the blocks of information displayed becomes too large, rapid skimming of the page can be impeded.
31.	Selecting appropriate page lengths The length of a page should be selected so as to support the primary purpose and use of the page. Short pages are generally more appropriate for homepages, navigation pages, or overview pages that need to be read quickly. Longer pages can be more appropriate when users want to read the content without interruptions or when the page needs to match a paper counterpart.
32.	Using colour: Colour should be used with care, taking into account human capabilities and restrictions in perceiving colour, and not as the only means of conveying information. Color should never be the only means of coding. Some users may have difficulties in perceiving certain colors or color combinations

	(color-blindness).
33.	Using frames with care: If frames are used, care should be taken to avoid possible problems, for example, those involving the use of the back button, bookmarking of pages, or scrolling of information. When frames are used, it is important to title each frame, and to describe its purpose and how frames relate to one another other.
34.	Providing alternatives to frame-based presentation: If frames are used, an alternative way of presenting relevant information without frames should be provided. Providing alternative text-only pages: When style sheets and/or frames are turned off it should be possible for the user to read and understand the page; alternatively, the user should be provided with an equivalent alternative text-only page.
35.	Consistent page layout: Pages should be designed using consistent layout schemes, supporting the user in finding similar information at the same position on different pages. Overall layout schemes apply to all pages and are preferable when all pages have a similar structure. Frequently, however, different pages have different purposes and types of content. In such cases, pages can usually be grouped in different categories, using one layout scheme for each category consistently.
36.	Placing title information consistently: Page titles should be placed in a consistent location on the different pages.
37.	Observing principles of human perception When designing application pages, the general principles of human perception should be taken into account. The International Standards mentioned below shall be consulted for guidance. Practical guidelines for presenting information to the user are to be found in ISO 9241-12. Guidance on selecting and using different forms of interaction techniques is to be found in ISO 9241-14 to ISO 9241-17. ISO 9241-14 gives guidance about menus, ISO 9241-15 about command dialogues, ISO 9241-16 about direct manipulation and ISO 9241-17 about forms. In addition, when designing multimedia information presentations, the design principles and recommendations described in ISO 14915-1 to ISO 14915-3 should be taken into account. Appropriate content presentation also plays a key role in accessibility.
38.	Linking back to the home page or landmark pages: Each page should contain a link leading to the home page of the application or to a landmark page that is easy to recognize for the user.
39.	Providing a site map: A separate navigation overview such as a site map should be provided for application showing the structure of the site in an overview form.
40.	Consistency between navigation components and content: If navigation components (or overviews) are shown in conjunction with associated content, consistency between the navigation component and the content shown should be maintained by indicating in the navigation component (e.g. highlighting) the topic

	currently visible in the content area.
41.	Placing navigation components consistently: Navigation components should be placed consistently on the pages or in the framesets in the pages of the application.
42.	Individualization and user adaptation : Adapting the content and the navigation of a user interface to individual users or user groups can be a useful mechanism for providing information that is of interest to the users and for making access to relevant information more efficient. User adaptation can also be important for making the user interface more accessible. Different approaches can be used for achieving these goals, like providing users with means for customizing the user interface to their personal needs i.e. individualization designing content and navigation differently for varying user groups or roles i.e. such as employees of different levels, citizens etc, monitoring the user's behaviour and adapting to the user's goals that are inferred from the behaviour observed, recommending information that is potentially more relevant or interesting to the specific user, based on the behaviour of all users or a user group.
43.	Taking account of the users' tasks and information needs: When providing different access paths or navigation structures for different user groups, the tasks and information needs of these user groups should be taken into consideration.
44.	Making individualization and adaptation evident: It should be made evident to the user when individualization and/or adaptation are used.
45.	Making user profiles evident: If predefined user profiles or user-specified profiles are used for individualizing or adapting content, the profile currently used should be made evident. If profiles are used, it is important to provide users with information about this concept and its implications.
46.	Allowing users to see and change profiles: If user-specified profiles are used, users should be able to see, modify and delete that profile on demand.
47.	The user interfaces of the system should follow the guidelines specified under www.usability.gov

| System Availability	
1.	The System must be available to users: • from <xx:00> to <xx:00>; • on <all weekdays/xxx days per year>.
2.	The planned downtime for the System must not exceed <xx> hours per <rolling three month period>.

	The System is considered to be down if any user is unable to perform any normal System function and if this failure is attributed to any component of the System other than the workstation.
3.	Unplanned downtime for the System must not exceed <xx hours/minutes> per <rolling three month period>.
4.	The number of incidents of unplanned downtime for the System must not exceed <x> per <rolling three month period>.
5.	In the event of any software or hardware failure, it must be possible to restore the System (with inline synchronization) within no more than <xx> hours.

Performance and Scalability	
1.	The System must provide adequate response times for commonly performed functions under both standard and peak conditions
2.	The System must be able to perform a simple search within 5-8 seconds and a advanced search (multiple search criteria) within 10-15 seconds regardless of the storage capacity or number of cases in the system. In this context, performing a search means returning a result list. It does not include retrieving the records themselves.
3.	The System must be able to retrieve and display within 5-8 seconds the case which has been accessed within the previous 2 months, regardless of storage capacity or number of cases in the system. This requirement is intended to allow for rapid retrieval of frequently-used cases, on the understanding that frequency of use is typically correlated with recent use.
4.	The System must be able to retrieve and display within 20 seconds the case which has not been accessed within the previous 2 months, regardless of storage capacity or number of cases in the system. This requirement is intended to allow for cases where cases used infrequently are stored on slower media than more active records.
5.	The System be scaleable and must not have any features which would preclude use in small or large police stations, with varying numbers of cases handled.

5. FUNCTIONAL ARCHITECTURE RECOMMENDATIONS

The proposed functional architecture is modeled around centralized deployment to facilitate ease of maintenance and leverage advancement in open standards and web technologies.

The 3 C's (Core-Configuration-Customization) forms the guiding principle for the architecture. The functional architecture of the CCTNS solution is given in the figure below. The functional architecture is composed of 4 major components based on SOA principles. Each of the components contains multiple services as defined by Service Definition. The core services, support layer and security and access control components can be deployed as standard components with necessary configuration changes. The customization layer can override and add to the core services based on the specific state requirements and can be plugged with the core services.
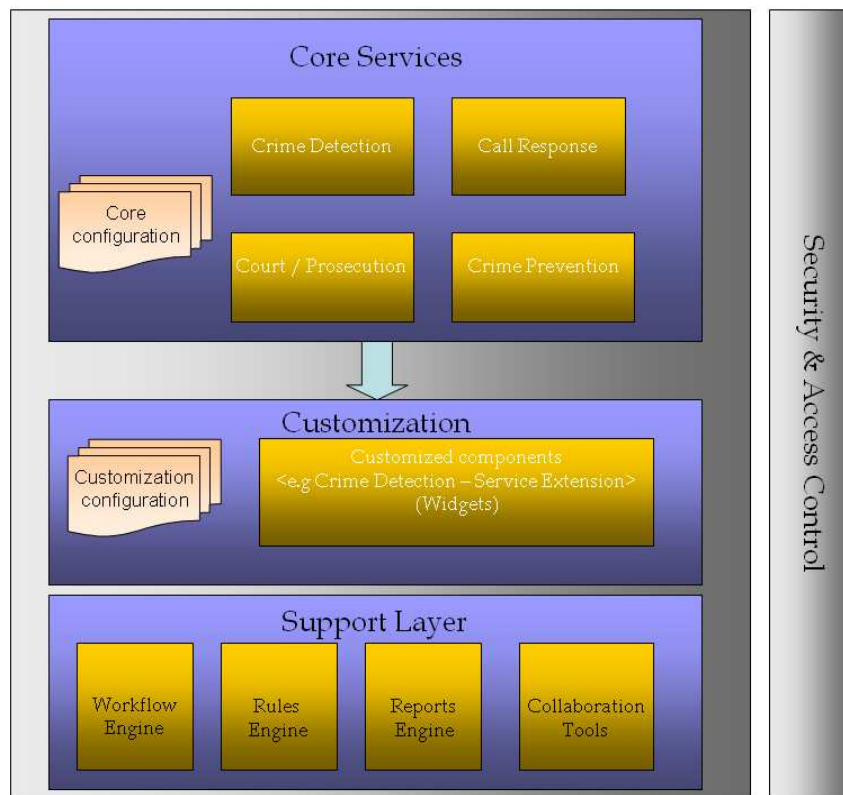
CCTNS Functional Architecture Overview

The deployment of the application will be at state level and will be configured and customized as per the state specific extensions.

The systems should be designed with the following broad guidelines:

	System Functionality
1)	The system should support multilingual interface
2)	The system should be designed in manner that operational data is not lost in case of any failure of equipment or communication network.
3)	The system should work even in an offline mode with the critical functionality
4)	The system should be designed to have satisfactory performance even in Police Stations connected on low-bandwidth
5)	The system should be implemented using Service Oriented Architecture (SOA) and have a modular design
6)	The system should be developed on Open Standards
7)	The system should be built on a common User Access and Authentication Service to ensure Single-Sign on for the end-user
8)	The system should be developed for a centralized deployment and maintenance
9)	The system should be developed to be deployed in a 3-tier datacenter architecture
10)	The system should be designed to have a n-tier architecture with the presentation logic separated from the business logic that is again separated from the data-access logic
11)	The system should be extensible to provide access to the interfaces through PDA's and mobile data terminals
12)	The system should adopt standardized formats and common metadata elements
13)	The system should be designed for access through browser-based systems and must impose minimal requirements on the client device
14)	The system must support multiple types of communication services for remote access
15)	The system should have capability to support public access to a subset of data and functionality
16)	The system should support multi-tier authentication where required

17)	The system should support SSL encrypted connections
18)	The system should support secure virtual private network connections
19)	The system should use HTTPS as the communication protocol, i.e., HTTP over an encrypted secure socket layer (SSL)
20)	The system should run on multiple browsers
21)	The system should support selective encryption of the stored data
22)	The system should ensure secure transmission of data over the network and utilize SSL and 2-way digital signatures
23)	The system should ensure high standards of security and access control through: a) *Prevent cross-site scripting* b) *Validate the incoming data / user request* c) *Encode the incoming data / user request* d) *Prevent SQL Injection* e) *Utilize parameterized queries* f) *Sanitize the user-inputs* g) *Validate the data both at the client and server* h) *Do not allow hard delete and perform only soft tagging the row for deletion*
24)	The system should ensure high scalability and performance through: a) *Use of cache for storing frequent data* b) *Use of AJAX based technology to improve user experience. Aggressive page loading to be considered based on the screen and estimate usage pattern* c) *Leverage Asynchronous HTTP socket capabilities of web server for scalability and performance* d) *Host all the static content (documents, images) on the web server* e) *The search results should be fetched from the database in batches of 10 or 20 maximum as configured within the application* f) *Display of records on the screen in batches/paged manner* g) *The search should fetch only the fields that need to be displayed to the user. Only when the user clicks on a particular record to view its further details should a query be fired to fetch the additional details for this particular record only* h) *A hierarchical cache should be configured and used for caching of results of most*

	frequently used searches
	i) *Database Indexes should be applied on the key columns used for searching*