Judiciary Information System

Software Requirement Specification

Version 3.0

Group-14

Ashutosh Dash   122CS0074

Sourav Ranjan Nayak 122CS0094

Rahul Swarnkar   122CS0114

Revision History

Date	Version	Description	Author
05/02/2025	1.0	SRS 1.0	Group 14
12/02/2025	2.0	SRS 2.0	Group 14
08/03/2025	3.0	SRS 3.0	Group 14

Table of Contents

Problem Statement

1. Judiciary Information System

The attorney general's office has requested us to develop a Judiciary Information System (JIS), to help handle court cases and also to make the past court cases easily accessible to the lawyers and judges. For each court case, the name of the defendant, defendant's address, the crime type (e.g., theft, arson, etc.), when committed (date), where committed (location), name of the arresting officer, and the date of the arrest are entered by the court registrar. Each court case is identified by a unique case identification number (CIN) which is generated by the computer. The registrar assigns a date of hearing for each case. For this the registrar expects the computer to display the vacant slots on any working day during which the case can be scheduled. Each time a case is adjourned, the reason for adjournment is entered by the registrar and he assigns a new hearing date. If hearing takes place on any day for a case, the registrar enters the summary of the court proceedings and assigns a new hearing date. Also, on completion of a court case, the summary of the judgment is recorded and the case is closed but the details of the case is maintained for future reference. Other data maintained about a case include the name of the presiding judge, the public prosecutor, the starting date, and the expected completion date of a trial. The judges should be able to browse through the old cases for guidance on their judgment. The lawyers should also be permitted to browse old cases, but should be charged for each old case they browse.

Using the JIS software, the Registrar of the court should be able to query the following:

(a) The currently pending court cases. In response to this query, the computer should print out the pending cases sorted by CIN. For each pending case, the following data should be listed: the date in which the case started, the defendant's name, address, crime details, the lawyer's name, the public prosecutor's name, and the attending judge's name.

(b) The cases that have been resolved over any given period. The output in this case should chronologically list the starting date of the case, the CIN, the date on which the judgment was delivered, the name of the attending judge, and the judgment summary.

(c) The cases that are coming up for hearing on a particular date.

(d) The status of any particular case (cases are identified by CIN).

The lawyers and the judges need to refer to the past court cases. The lawyers need to refer these to prepare for their line of arguments. The judges need to refer the past court cases to examine the lines of judgments given previously to similar cases. It should be possible to search for the history of past court cases by entering key words. However, the lawyers should be charged for each time they see the details of a court case to recover some of the computerization costs. For this purpose, it is necessary to provide separate login accounts to the JIS software and keep track of how many court cases each lawyer views. The registrar should be able to create login accounts for the different users (i.e. judges, lawyers, etc) and should be able to delete these accounts.

Software Requirement Specification

2. Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **Judiciary Information System** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the **Judiciary Information System** are provided in this document.

2.1. Purpose

The purpose of this document is to provide detailed information regarding Judiciary Information System. It will explain the features of the system, constraints under which it will operate and how the system will react to external stimuli. This document is intended to all the stakeholders and developers of the system.

2.2. Scope

This application will be designed to provide a helping hand for the judges as well as the lawyers of the court by providing tools to help handle court cases which would otherwise have to be performed manually.

- This system will make the past court cases easily accessible to the lawyers and judges.
- The judges would be able to browse through the old cases for guidance on their judgement and examining the lines of judgement given previously to similar cases.
- It would be possible to search for the history of past court cases by entering CIN.
- The lawyers would be permitted to browse old cases, but would be charged for each old case they browse.
- For the purpose of maintaining the charge for lawyers, separate login accounts are created for the lawyers and the judges which are maintained by the Registrar of the court.
- This system will allow the Registrar to see the details of the currently pending cases and the cases which have been resolved or the status of any particular case.

2.3. Definitions, Acronyms, and Abbreviations

JIS-Judiciary Information System

CIN-Case Identification Number

3. Functional Requirements

3.1. R1: Logging in of Registrar

Brief Description: The Registrar logs in the system and Inputs the details of the case.

Input: The Registrar logs into the system by selecting the **Registrar Log-In** option. The defendant's name, defendant's address, crime type, date of crime, place of crime, name of arresting officer and the date of arrest for each case are entered by selecting the **Input Case Details** option.

Processing: The application stores the log-in details of the users and matches it against the Input.

Output: The computer automatically generates a unique case identification number (CIN) for each case.

3.2. R2: Assigning Hearing Date

Brief Description: After the unique CIN is generated, the Registrar assigns a date of hearing for the case.

Input: The Registrar selects the **Display Dates** option.

Processing: The application checks if they are occupied or not and prints the non-occupied dates.

Output: The computer displays the vacant slots on any working day during which the case can be scheduled.

3.3. R3: Adjourning a Case

Brief Description: Reason of adjournment is entered if any case is adjourned.

Input: The Registrar enters the reason due to which the case was adjourned by selecting **Enter Summary** option and selects the **Display Dates** option.

Processing: The application updates the summary and stores it in the database.

Output: A new hearing date is assigned for that case.

3.4. R4: Showing Pending Court Cases

Brief Description: This function gives the details of the currently pending cases when queried by the Registrar.

Input: The Registrar queries about the pending cases by selecting the **Pending Cases** option.

Processing: The application filters out the pending court cases

Output: The application shows the pending court cases. For each pending case, the following data are listed: the date in which the case started, the defendant's name, address, crime details, the lawyer's name, the public prosecutor's name and the attending judge's name.

3.5. R5: Showing Resolved Cases

Brief Description: This function displays the details of the resolved cases over any given period.

Input: The Registrar queries about the resolved cases by selecting the **Resolved Cases** option.

Processing: The application filters out the resolved court cases

Output: The application shows resolved court cases, the CIN, the date on which the judgement was delivered, the name of the attending judge and the judgement summary.

3.6. R6: Showing Cases on a particular date

Brief Description: This function lists the cases due on a particular date.

Input: The Registrar selects the **Due Cases** option and enters the date of hearing.

Processing: The application filters out cases on the basis of date.

Output: All the cases that are scheduled on that day are listed in the form of their CIN.

3.7. R7: Showing Case Status

Brief Description: This function displays the status (Pending/Closed) of any particular case queried by the Registrar.

Input: The Registrar selects the **Case Status** option and enters the CIN of the case he is interested in.

Processing: The application filters out on the basis of status and CIN.

Output: The computer displays the status of the particular case.

3.8. R8: Create/Delete Accounts

Brief Description: This function allows the Registrar to create or delete judges/lawyers accounts.

Input: The Registrar creates accounts by selecting the **Create New Account** option and entering the username of the judge/lawyer.

He deletes an account by selecting the **Delete Account** option and entering the username of the judge/lawyer.

Processing: The application creates/deletes the judge/lawyer.

Output: A username and password is created for every account created and deleted for every account deleted.

3.9. R9: Logging in Judges

Brief Description: This function allows the judges to log into the JIS and browse through the previous case history to get guidance on their decisions.

Input: The judges log into the system by selection the **Judges Log-In** option and can select the previous cases by selecting the **Resolved Cases** option and entering key words like their CIN.

Processing: The application logs in the judge.

Output: The case details of the particular case are displayed.

3.10. R10: Logging in Lawyers

Brief Description:

This function allows the lawyers to log into the JIS and browse through the previous case history to get guidance on similar cases.

Input: The lawyers log into the system by selection the **Lawyers Log-In** option and can select the previous cases by selecting **Resolved Cases** option and entering key words like their CIN.

Processing: The application logs in the lawyer.

Output: The case details of the particular case are displayed. Also, the number of previous cases views for each lawyer is displayed.

3.11. R11: Pay Charge

Brief Description:

This function allows the lawyers to clear their dues for viewing previous court cases.

Input: The lawyers can pay for their charges by logging into JIS and selecting **Pay Charges.**

Processing: The application updates the charges against the lawyer.

Output: This connects the JIS to the Billing System which generates the printed bill and resets the charges to NIL for the lawyer.

4. External User Interface Requirements

The user interface is basically divided into three main sections: the interface related to the Registrar, the judge and the lawyer and are as follows:

4.1. Registrar

The Registrar interface allows logging in, entry of case details, scheduling dates foe cases, summarizing and adjourning of cases, checking case status, creating/deleting judges and lawyers, and log-out to the homepage.

4.2. Judge

The Judge interface allows logging in, viewing resolved case-details, and log-out to the homepage.

4.3. Lawyer

The Lawyer interface allows logging in, viewing resolved case-details and paying the charge for it, and log-out to the homepage.

5. Non – functional Requirements

5.1. Portability

The software should be compatible with all the main stream Web browsers. The user can log in to the system at any time.

5.2 Reliability

The system can be accessed by multiple users concurrently. Any user can access the system by using even a low performance PC.

5.3. Business Rules

The software will be free to use for all three users (Registrars, Lawyers and Judges). The source code will be publicly hosted for free use, modification and to get any useful advice to enhance our system. Users shall be able to contact administrators using emails.

6. Data Dictionary

6.1 Registrar Table

Field Name	Data Type	Description
Registrar ID	Integer	Unique identifier for the registrar (e.g. 10).
Registrar Name	String	Full name of the registrar.
Username	String (Unique)	Username for logging into the JIS system.
Password	String	Hashed password for secure login.

Email	String	Email address of the registrar for communication.
Contact Number	String	Contact number of the registrar.
Total Cases Registered	Integer	Total number of cases registered by the registrar.
Total Accounts Created	Integer	Total number of user accounts (judges/lawyers) created by the registrar.
Total Accounts Deleted	Integer	Total number of user accounts (judges/lawyers) deleted by the registrar.

6.2 Judge Table

Field Name	Data Type	Description
Judge ID	Integer	Unique identifier for the judge (e.g. 4)
Judge Name	String	Full name of the judge.
Username	String (Unique)	Username for logging into the JIS system.
Password	String	Hashed password for secure login.
Email	String	Email address of the judge for communication.
Contact Number	String	Contact number of the judge.
Assigned Cases	List of CINs	List of Case Identification Numbers (CINs) assigned to the judge.
Total Cases Resolved	Integer	Total number of cases resolved by the judge.
Specialization	String	Area of expertise or specialization (e.g., Criminal, Civil, Family Law).

6.3 Lawyer Table

Field Name	Data Type	Description
Lawyer ID	Integer	Unique identifier for the lawyer (e.g. 21)
Lawyer Name	String	Full name of the lawyer.

Username	String (Unique)	Username for logging into the JIS system.
Password	String	Hashed password for secure login.
Email	String	Email address of the lawyer for communication.
Contact Number	String	Contact number of the lawyer.
Cases Viewed	List of CINs	List of Case Identification Numbers (CINs) viewed by the lawyer.
Total Charges	Decimal	Total charges incurred by the lawyer for viewing past cases.
Specialization	String	Area of expertise or specialization (e.g., Criminal, Civil, Family Law).

6.4 Case Details Table

Field Name	Data Type	Description
Case Identification Number	Integer	Unique Identifier for the case.
Defendant Name	String	Full name of the Defendant.
Defendant Address	String	Complete Address of the Defendant.
Crime Type	String	Type of the Crime Committed
Crime Date	Date	Date when the Crime was Committed.
Crime Location	String	Location where the Crime was Committed.
Arresting Officer	String	Name of the officer who made the arrest
Arrest Date	Date	Date of arrest.
Assigned Judge ID	Integer (FK)	ID of the judge assigned to the case.
Lawyer ID	Integer (FK)	ID of the defense lawyer.
Case Start Date	Date	The date when the case was initiated.
Case Status	ENUM ('Pending', 'Resolved', 'Adjourned')	Current status of the case.

6.5 Hearings Table

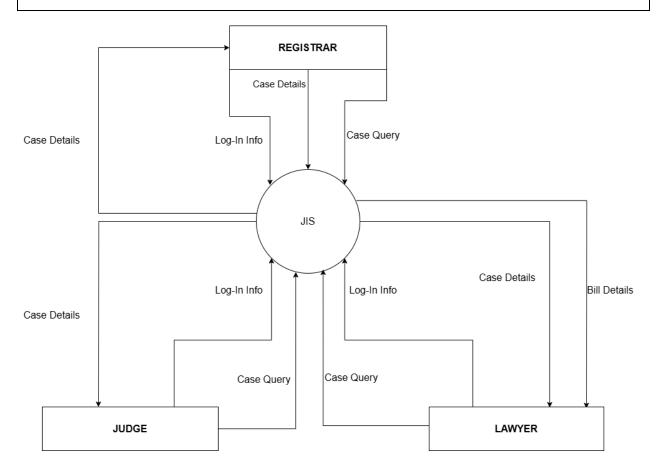| Field Name	Data Type	Description
Hearing ID	Integer(FK)	Unique identifier for each hearing.
Case ID	Integer(FK)	Case associated with the hearing.
Hearing Date	Date	Date of the hearing.
Adjournment Reason	String	Reason for adjournment, if applicable.
Next Hearing Date	Date	Date for the next Hearing.

6.6 Billing Table

Field Name	Data Type	Description
Bill ID	Integer	Unique billing record identifier.
Lawyer ID	Integer (FK)	Lawyer who viewed a case.
Case ID	Integer (FK)	Case viewed.
View Date	Timestamp	Date and time when the case was viewed.
Charge Amount	Decimal(10,2)	Charge for viewing the case.
Payment Status	ENUM ('Pending', 'Paid')	Status of the payment.

7. Data Flow Diagrams

7.1 Level 0 DFD

7.2 Level 1 DFD

8. Structure Chart

9. Use Case Diagram

9.1 Register Cases

9.2 Query Cases

10. Class Diagram

Users

-Name: String
-Username: String
-Password: String
-Email: String
-Contact Number: String

+login(): Void

Lawyer

-LawyerID: Integer
-Charges: Decimal
-Specialization: String

+getResolvedCases(): Cases
+getCaseDetails(): Case
+payCharge(): void
+viewBill(): Bill

Registrar

-registrarID: Integer
-totalcasesRegisterd: Integer
-totalAccountsCreated: Integer
-totalAccountsDeleted: Integer

+registerCases(): String
+adjournCases(): void
+updateCaseDetails():void
+inputCaseDetails():void
+getCaseDetails(): Case
+getResolvedCases(): Cases
+getPendingCases(): Cases
+selectDate(): Date

Judges

-JudgeID: Integer
-TotalCaseResolved: Integer
-AssignedCases: List
-Specialization: String

+getResolvedCases(): Cases
+getCaseDetails(): Case

Bill

-BillID: Integer
-LawyerID: Integer
-CaseID: Integer
-ViewDate: Date
-PaymentStatus: boolean

+viewBill(): Bill

Cases

-DefendantName: String
-DefendantAddress: String
-CrimeType: String
-CrimeDate: String
-CrimeLocation: String
-OfficerName: String
-ArrestDate: String
-LawyerID: Integer
-JudgeID: Integer
-ProsecutorID: Integer
-JudgementDate: String
-StartingDate: String
-CaseStatus: String
-CaseSummary: String

+getDefendantName(): String
+getDefendantAddress(): String
+getCrimeType(): String
+getCrimeDate(): String
+getCrimeLocation(): String
+getOfficerName(): String
+getArrestDate(): String
+getLawyerName(): String
+getJudgeName(): String
+getProsecuterName(): String
+getJudgementDate(): String
+getStartingDate(): String
+getCaseStatus(): String
+getCaseSummary(): String

Hearing Details

-HearingID: Date
-CIN: Integer
-date: Date
-isEmpty: Boolean

+showEmptyDates(): Date
+chooseFinalDate(): Date

11. Activity Diagram

11.1 Registrar's Activity Diagram

11.2 Activity Diagram for Quering Cases and Billing

12. ER Diagram