

Software Requirements Specification

for

Judiciary Information System

Version 1.0 approved

Prepared by:

Amartya Mandal Divyansh Bhatia Esha Manideep

Team: The Normies

Group No. : 39

18th March, 2021

Table of Contents

Table of Contents.....	2
1. Introduction	3
1.1 Purpose	3
1.2 Document Conventions	3
1.3 Intended Audience and Reading Suggestions	3
1.4 Product Scope	3
1.5 References	4
2. Overall Description	4
2.1 Product Perspective	4
2.2 Product Functions	4
2.3 User Classes and Characteristics	5
2.4 Operating Environment	5
2.5 Design and Implementation Constraints	5
2.6 User Documentation	5
2.7 Assumptions and Dependencies	5
3. External Interface Requirements	6
3.1 User Interfaces	6
3.2 Hardware Interfaces	7
3.3 Software Interfaces	7
3.4 Communications Interfaces	7
4. System Features	7
1.1 User Login	7
1.2 User Logout	8
1.3 Query Case	8
1.4 Create Case	9
1.5 Update Case	9
1.6 Create User	10
1.7 Delete User	10
1.8 Browse Case	11
1.9 Add Amount	11
5. Other Nonfunctional Requirements	12
5.1 Performance Requirements	12
5.2 Safety Requirements	12
5.3 Security Requirements	12
5.4 Software Quality Attributes	12
5.5 Business Rules	12
6. Other Requirements.....	13
Appendix A: Glossary.....	13
Appendix B: Analysis Models.....	13
Appendix C: To Be Determined List.....	13
Revision History.....	14

1. Introduction

1.1 Purpose

The main motive behind this document is to present a detailed description of the Judiciary Information System. This document describes in substantial detail, the software requirements of the Judiciary Information System, which is a software designed to handle court cases with increased efficiency and greater ease. It will provide a detailed explanation of the purpose and features of the system, information about the interfaces of the system, the conditions and constraints for its operation and how the system will react to external input given by the user. This will be proposed to the office of the Attorney General for its approval. The developers should consult this document as the only source of requirements for the project.

1.2 Document Conventions

- *Entire Document is written in Times New Roman font*
- *Main Headings are numbered with whole numbers like **1. Introduction**, written in bold*
- *Subheadings are numbered with decimal numbers like **1.1 Purpose**, written in bold*
- *Rest of the Document is written in Italic*
- *Font size: Main Heading (18), Subheadings(14), Rest of Document (11)*
- *Bullet points*
- *JIS stands for Judiciary Information System*

Headings are Bold so that we can easily distinguish them from the rest of the document. Bullet use to identify special information about the system.

1.3 Intended Audience and Reading Suggestions

This is a domain-specific technical document, intended for all the stakeholder customers and the developers of the system (designers, testers, maintainers) to assist in the development process of JIS as well as to serve a reference to clarify any future issues that the stakeholders may run into.

The reader is assumed to have basic knowledge of operating systems, databases and user accounts. Readers should have the knowledge of UML diagrams so that they can understand them.

1.4 Product Scope

The aim of the Judiciary Information System is to handle court cases by maintaining a digital record of everything related to court cases and to help lawyers and judges to access the past cases information easily.

- *The Court Registrar is the administrator of the system and can register new cases, assign a hearing date, maintain its detail and will save its information as past records when closed.*
- *Lawyers and Judges can only browse through past cases without changing or modifying any details.*

The JIS will have a login - based common user interface through which the different users (Registrars, Lawyers and Judges) can access their respective accounts and perform respective functionalities.

1.5 References

The following were referred to while making the SRS document together with the one already given.

- 1) IEEE. IEEE Std 830–1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
- 2) <https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>

2. Overall Description

2.1 Product Perspective

The Judicial Information System Software (JISS) is a program to be used by the judges, lawyers and registrars to improve the efficiency in handling the court cases. JISS makes cases easy to handle for the registrars by making it easy to assign dates to each case, search for pending cases on a given date etc.. rather than going through a lot of physical documents to do the same. System also provides information related to the cases which have been resolved so that the judges can refer to them to improve their judgements and so that the lawyers can develop their line of arguments. JISS aims to improve the case handling system of courts by replacing physical documents and papers, which can sometimes be tedious to handle.

2.2 Product Functions

The software is made to help Judges/Lawyers to access the old cases, and to make it easier for registrars in assigning hearing dates. The functions of the system include the system providing different types of services based on the type of users [Registrar/Judge/Lawyer].

- *Judges will be able to browse through their old cases for guidance on their judgement and examine the lines of judgement given previously to similar cases.*
- *Lawyers are also allowed to browse through old cases for preparing their line of arguments but will be charged for each old case they access.*
- *Lawyers have to keep some money in their accounts. Once they access a case, money gets deducted from their wallet. Once they can't browse cases anymore(when they are out of money in their wallet), they need to pay the registrar to add money to their respective wallets.*
- *The registrar can assign a date of hearing for each case. System also displays the vacant slots of any working day.*
- *The registrar can get the information about the currently pending cases, the cases which have been resolved, the cases that are coming up for hearing on a particular day and the status of any particular*

case.

- *Registrar can add/delete accounts of judges/lawyers.*
- *Search for old cases is done by entering a keyword.*

2.3 User Classes and Characteristics

There are three main types of users. They are:

- *Judges*
- *Lawyers*
- *Registrar*

Registrar is an admin for the whole system. All the users are assumed to have basic knowledge of computers, internet, how to search etc. Registrar is supposed to know a little more about computers, just in the rare case of disk crashes, power failures etc.

2.4 Operating Environment

- *System is supposed to have an internet connection to connect to the server.*
- *Software works on all OS(Windows/Linux/MacOS)*
- *The System must have a web browser installed.*

2.5 Design and Implementation Constraints

- *The files in which the information regarding the previous cases are stored should be secured against malicious deformations*
- *Users are supposed to remember their password and username as there is no 'forgot username/password' option.*
- *Lawyers must make sure that the money they paid the registrar is reflected in their specific account wallets.*

2.6 User Documentation

Brief description of user interfaces and a user's manual are sufficient to understand how to use the system. Maintenance guide is provided to the Registrar along with some other description of some other interfaces(add/delete accounts).

2.7 Assumptions and Dependencies

- *Users are supposed to have very basic knowledge of computers etc(using mouse, keyboard, clicks etc)*
- *The whole interface is provided in english, so all the users are assumed to be decently fluent in english*
- *Old cases that were closed before the release of the software are supposed to be uploaded into JISS once it is installed by the registrar so that the users can access older cases too.*

3. External Interface Requirements

3.1 User Interfaces

The user shall follow basic windows style and functionality conventions. The user interface is divided into three sections corresponding to the interfaces related to the Registrar, the judge and the lawyers :

1. Registrar

- a. **Log-in** : There is a Login button placed on the homepage of the JIS software. The Registrar logs into the system by entering his username and password in the respective places. In case of wrong or unknown username or password, an error will be shown on clicking the log-in button.*
- b. **Create Case** : There is a Create Case button placed on the Registrar's interface of the JIS software. The registrar enters the details of a new case into the system.*
- c. **Query Case** : There is a Query Case button placed on the Registrar's interface of the JIS software. Once the Registrar logs into the system, he can query for pending, resolved, upcoming or a specific case status as required.*
- d. **Update Case** : There is a Update Case button placed on the Registrar's interface of the JIS software. The Registrar can update the case after a hearing, adjourn a case, or can close a case.*
- e. **Create User** : There is a Create User button placed on the Registrar's interface of the JIS software. The Registrar has the ability to create new users. The Registrar can add a new Judge or a new Lawyer as required.*
- f. **Delete User** : There is a Delete User button placed on the Registrar's interface of the JIS software. The Registrar has the ability to delete an existing user of a Judge or a Lawyer as required.*
- g. **Add Amount** : There is a Add Amount button placed on the Registrar's interface of the JIS software. The Registrar has the power to add money to the Lawyer's wallet on payment for the same.*
- h. **Log-out** : There is a Log-out button placed on the Registrar's interface of the JIS software. On selecting this button, the Registrar logs out of the system and the homepage of the software is displayed.*

2. Lawyer

- a. **Log-in** : There is a Login button placed on the homepage of the JIS software. The Lawyer logs into the system by entering his username and password in the respective places. In case of wrong or unknown username or password, an error will be shown on clicking the log-in button.*
- b. **Browse Case** : There is a Browse case button placed on the Lawyer's interface of the JIS software. The Lawyer has the ability to browse past cases to prepare their line of argument, by paying some charge.*
- c. **Log-out** : There is a Log-out button placed on the Lawyer's interface of the JIS software. On selecting this button, the Lawyer logs out of the system and the homepage of the software is displayed.*

3. Judge

- a. **Log-in** : There is a Login button placed on the homepage of the JIS software. The Judge*

logs into the system by entering his username and password in the respective places. In case of wrong or unknown username or password, an error will be shown on clicking the log-in button.

- b. **Browse Case :** There is a Browse case button placed on the Judge's interface of the JIS software. The Judge can browse past cases to examine the line of judgement in similar cases.
- c. **Log-out :** There is a Log-out button placed on the Judge's interface of the JIS software. On selecting this button, the Judge logs out of the system and the homepage of the software is displayed.

3.2 Hardware Interfaces

The Judiciary Information System uses hardware similar to most online web applications including a monitor, keyboard and a mouse/trackpad. Monitor is used for display purpose, mouse to control navigation and keyboard to control user input. A decent internet connection is also required. The supported devices are computers on any operating system as long as there is modern web browsing functionality.

It uses a computer database with a network server. The user's computer transfers and receives data from the server using basic networking protocols. All system's information is stored in the server's database. The server is directly connected to the client systems. The client systems have access to the database in the server.

3.3 Software Interfaces

The Application should be able support all different web browsers available so that the user can access our system with ease. The system will include a database (SQL server) to store information regarding cases or user credentials. The database will be secured with a username-password combination. The Back-end, or the database services will be used greatly, thus requiring quite an efficient design. A firewall will be used with the server to prevent unauthorized access to the system.

3.4 Communications Interfaces

Users may email any bugs they encounter while using to address issues with the software. This system supports all types of web browsers. The system will follow secure user logins to protect user information. The system follows HTTP communication standards and uses simple electronic forms to show information.

4. System Features

4.1 User Login

4.1.1 Description and Priority

The Judiciary Information System handles login for Registrar, Judge and Lawyer such that they can operate without any interference from each other.

Registrar Login allows the user to query, create and update cases as well as add or remove

other users.

Judge Login allows the user to browse past cases.

Lawyer Login also allows the user to browse past cases for a charge.

4.1.2 Stimulus/Response Sequences

The Registrar can use Registrar Login for using the software.

The Judges can use Judge Login for using the software.

The Lawyers can use Lawyer Login for using the software.

4.1.3 Functional Requirements

SQL Database: If no database is present, then the user ids and passwords cannot be stored. Thus it is necessary to have a database at hand.

Flask: Without Flask, we cannot use Flask Login, which is used for Logging in a user.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

REQ-3: flask_req

4.2 User Logout

4.2.1 Description and Priority

The Judiciary Information System allows the Registrar, Judges and Lawyers to logout for security.

All users can safely logout after finishing their work to prevent any interference with other users.

4.2.2 Stimulus/Response Sequences

The Registrar can use Registrar Logout after using the software.

The Judges can use Judge Logout after using the software.

The Lawyers can use Lawyer Login after using the software.

4.2.3 Functional Requirements

Python: The programming language required to run the application

REQ-1: py_req

4.3 Query Case

4.3.1 Description and Priority

The Judiciary Information System handles queries from the Registrar.

The currently pending case details can be displayed.

Cases resolved in a particular period can be listed.

The upcoming case hearings can be displayed

Status of a particular case can be displayed by its CIN.

4.3.2 Stimulus/Response Sequences

The Registrar can query for pending, resolved, upcoming or a specific case status as required.

4.3.3 Functional Requirements

SQL Database: If no database is present, then the cases cannot be stored. Thus it is necessary to have a database at hand.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

4.4 Create Case

4.4.1 Description and Priority

The Judiciary Information System provides the ability to the Registrar to add a new case.

For adding a new case the Registrar enters the case details including 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. A unique CIN number and a hearing date is assigned to the case.

4.4.2 Stimulus/Response Sequences

The Registrar can update the case after a hearing, case adjournment, or case closure.

4.4.3 Functional Requirements

SQL Database: If no database is present, then the cases cannot be stored. Thus it is necessary to have a database at hand.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

4.5 Update Case

4.5.1 Description and Priority

The Judiciary Information System provides the ability to the Registrar to update cases.

Each time a case is adjourned, the reason for adjournment is entered and a new hearing date is assigned.

For every hearing, the summary is entered and a new hearing date is assigned.

On completion of a case, the summary is recorded and the case is closed.

4.5.2 Stimulus/Response Sequences

The Registrar can update the case after a hearing, case adjournment, or case closure.

4.5.3 Functional Requirements

SQL Database: If no database is present, then the cases cannot be stored. Thus it is necessary to have a database at hand.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

4.6 Create User

4.6.1 Description and Priority

The Judiciary Information System provides the ability to the Registrar to create new users.

The Registrar can add a new judge as required.

The Registrar can also add a lawyer as required.

4.6.2 Stimulus/Response Sequences

The Registrar can create a new user for a judge or lawyer.

4.6.3 Functional Requirements

SQL Database: If no database is present, then the user credentials cannot be stored. Thus it is necessary to have a database at hand.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

4.7 Delete User

4.7.1 Description and Priority

The Judiciary Information System provides the ability to the Registrar to delete existing users.

The Registrar can remove a judge when required.

The Registrar can also remove a lawyer when required.

4.7.2 Stimulus/Response Sequences

The Registrar can delete an existing user credentials for a Judge or a Lawyer.

4.7.3 Functional Requirements

SQL Database: If no database is present, then the user credentials cannot be stored. Thus it is necessary to have a database at hand.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

4.8 Browse Case

4.8.1 Description and Priority

The Judiciary Information System provides the ability to the Judges and Lawyers to browse past cases for reference.

Judges can refer to past cases to examine the line of judgement in similar cases.

Lawyers can refer to past cases to prepare their line of argument, by paying some charge.

4.8.2 Stimulus/Response Sequences

The Judges and Lawyers can browse past cases for reference.

4.8.3 Functional Requirements

SQL Database: If no database is present, then the cases cannot be stored. Thus it is necessary to have a database at hand.

Python: The programming language required to run the application

REQ-1: py_req

REQ-2: sql_req

4.9 Add Amount

4.9.1 Description and Priority

The Judiciary Information System provides the ability to the Registrar to add money to the wallet of a Lawyer on payment of the same.

This amount is used by the Lawyer to pay the charge for browsing past cases.

4.9.2 Stimulus/Response Sequences

The Registrar can add an amount to the wallet of a Lawyer.

4.9.3 Functional Requirements

Python: The programming language required to run the application

REQ-1: py_req

5. Other Nonfunctional Requirements

5.1 Performance Requirements

There are no specific constraints on the operating system as long as it has a decently recent version of any web browser. It must be able to perform in adverse conditions which may arise in case of slow internet connections and low battery on device.

5.2 Safety Requirements

Lawyers must ensure that they collect the bill from the registrar as soon as they add money to their wallets. All the users must remember their usernames and passwords as there is no facility of forgetting password or username.

5.3 Security Requirements

The database is protected with a username-password combination, thus allowing privacy. As all the details of cases stored must be secured from attacks, so that the database won't be corrupted. The system is secured in a way that each type of user is given specific permissions, thus preventing from interfering with each other's data.

5.4 Software Quality Attributes

- **Reliability** : *The system can be accessed by multiple users concurrently. Any user can access the system by using even a low performance PC*
- **Availability** : *The system will be available 24 hours of the day*
- **Maintainability** : *The system shall provide the capability to maintain the data*
- **Portability** : *The user can log in to the system at any time*
- **Robustness** : *The system will not be broken down easily and a single application failure will not affect the whole system.*

5.5 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. Other Requirements

Will be updated during development.

6.1 Attributes

- **Performance** : *Internet should be available all the time for the software to be able to perform efficiently. System should have enough space to be able to store all the data about all the cases.*
- **Portability** : *Use of latest versions of web browsers is recommended for best experience*
- **Maintainability** : *The user manuals and tutorials provided should be thoroughly studied so that there won't be any issues in the operating/maintaining the software.*

6.2 Database

- *Various files are required to maintain the details of the court cases and the users. Registrar must update the old cases (recently finished ones at least) so that the users can make better use of the software.*

6.3 Hardware

- *A dedicated server in the Attorney General's office for the functioning of the software. A printer near the registrar, so that it prints the bill whenever a lawyer needs to add money to his wallet so that he/she can access old case files for preparing their line of argument.*

Appendix A: Glossary

py_req: Python Required

sql_req: SQL Required

flsk_req: Flask Required

Database : A set of data in a structure of tables organised by rows and columns.

Server : A computer that manages documents and memory of a central source.

To be modified when more technical features are discussed and produced.

Appendix B: Analysis Models

Appendix C: To Be Determined List

Revision History

Name Date Reason For Changes Version			