Structured Analysis and Structured Design (SA/SD)

Judiciary Information System

14CS10059
Vaishal Shah

14CS30019
Mousam Roy

Table of Contents

1. Feasibility study

1.1 Understanding the problem

The Judiciary Information System is designed to provide a convenient platform to lawyers and judges to access data of old cases. This will provide them assistance while formulating their arguments and passing their judgments respectively. The Court registrar acts as the main administrator of the system and he has elevated privileges to create lawyer and judge user accounts. Apart from account management, he is also in charge of maintenance of the data of currently running cases, keeping them up to date with latest updates in the cases. When a case is closed in the Court, the registrar updates the corresponding information on the JIS and the case data is archived for later reference.

1.2 Scoping the problem

The scope of the problem and the various constraints it must work under are listed in the following points.

1. The JIS software would run on a standalone workstation.
2. All of the data will be maintained locally.
3. The users would have separate individual accounts accessible on providing corresponding user name and password. The user name and password for lawyers and judges will be set by the registrar. They may change their password later, but not their user name.
4. At most one user can use the system at a particular time.
5. At most one registrar account can be created. The registrar is free to choose his own user name and password. He may not change his user name.
6. No functionality for deletion of cases is to be present, since old cases are to be retained in the system for reference.

1.3 Analyzing stakeholders

The stakeholders for the JIS software are various people who can have accounts in the JIS. As mentioned before, they are:

1. **Registrar**: He acts the system administrator and has the below mentioned responsibilities and rights.

(A) *Documentation of cases*: create, adjourn, update and close

(B) *Querying cases*: by status (current, resolved), date of next hearing, CIN

(C) *Account management*: Create and delete lawyer and judge accounts

2. **Lawyer:** There may be more than one lawyer. His only function is to query cases by keyword and view case data. He is charged for every case viewed.

3. **Judge:** There may be more than one judge. His only function is to query cases by keyword and view case data.

1.4. Defining alternatives

Some of the below mentioned alternative features may be implemented in the system, though they are not expected to be present in the initial version of the software.

(A) Other methods of authentication such as fingerprint, One Time Password (OTP), security question etc may be implemented for advanced security.

(B) To model a more practical environment, lawyers and judges may be allowed to create accounts for themselves. However, they may not use their accounts (i.e. Browse cases) until their account is approved by the registrar.

(C) The databases may be moved to a web server instead of being kept locally. This would allow users to use the system simultaneously from multiple workstations.

1.5. Defining criteria to evaluate

- Cost of Technology
- Cost of Infrastructure
- Lifetime of Technology
- Stability of Technology

1.6. Assessment of unusual circumstances

1. **Password recovery**: In case a user (lawyer/judge) forgets his password, he needs to contact the registrar in person to retrieve his password. No support for password recovery or resetting will be implemented in the system for reasons of simplicity, though the system will be designed to facilitate addition of such a feature in a later version if required.

2. **Database out of memory**: For reasons of simplicity, we assume that this situation will not occur. However, any failure to register a new case in the system will be handled properly to ensure that the system does not crash and that no previously stored data is compromised.

3. **Cancellation:** Any ongoing process, if canceled, will safely return the system to a previous state without compromising the stability of the system or data integrity.

2. Requirements analysis

2.1. Functional requirements

2.1.1. Data flow

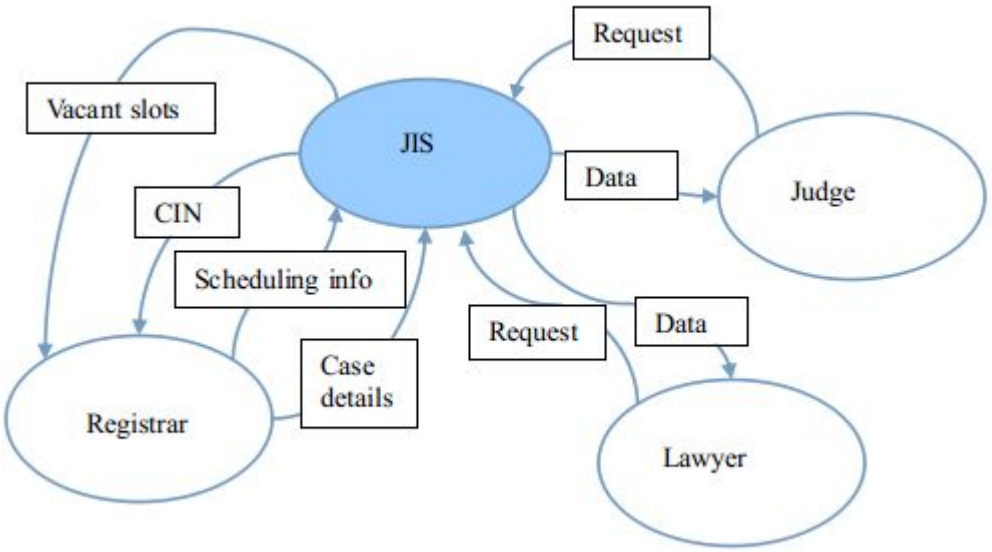

Fig 1: Context Diagram

The JIS is shown with the registrar, judge and lawyer as external entities.

Inputs: Case details, scheduling information, data requests
Process: Judiciary Information System
Outputs: CIN, vacant slots information, case data

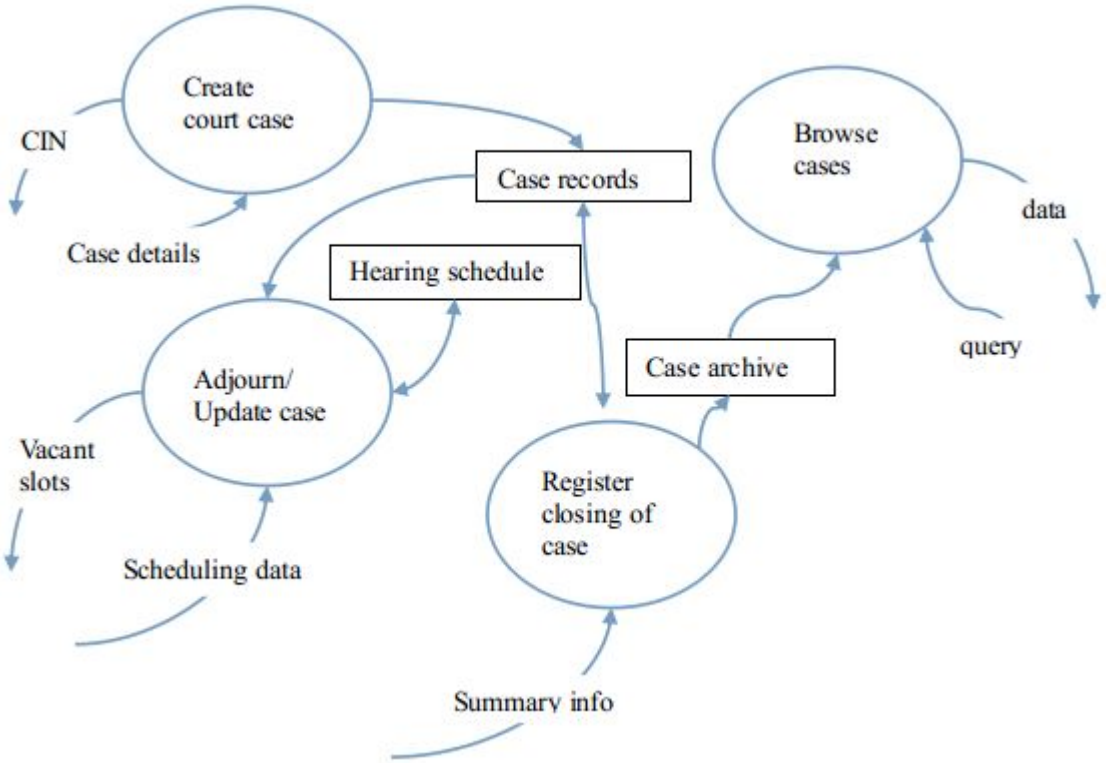

Fig 2: Data Flow Diagram (DFD)

2.1.2. Structure chart

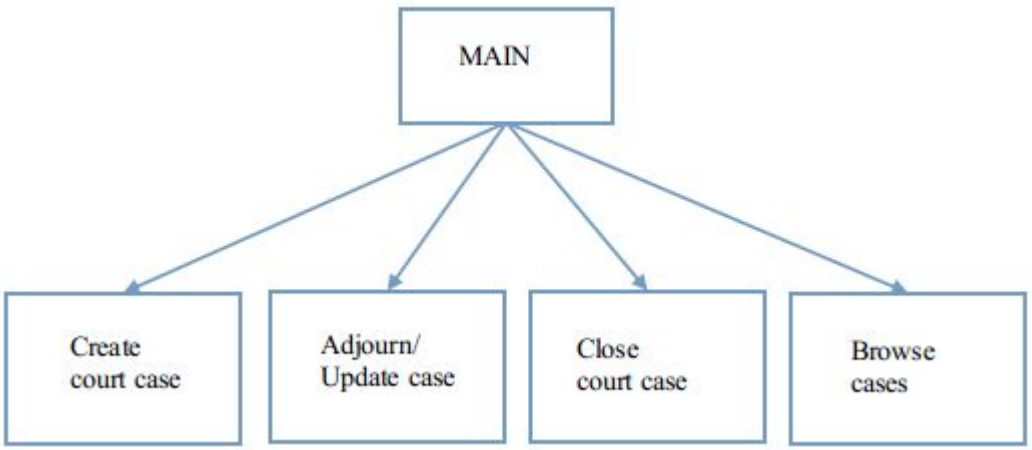

Fig 3: Structure Chart

2.2. Non-functional requirements

- *Database Requirements*: The size of database will increase proportionally to the number of case records. So, there must be space in hard disk to store the total amount of data which might be more than that specified in the hardware specifications

section.

- *Legal Requirements*: JIS cannot be distributed freely by anyone as it has a software license agreement.
- *Availability of JIS*: JIS will be available as long as the office of the Court is open and any of the stakeholders is present.

2.3. Report

In this section, the functional requirements of JIS are explained in detail using data flow diagrams and structure chart. The DFD graphically represents the "flow" of data through the system. The DFD along with Structure Chart help in visualization of data processing of JIS.
The non-functional requirements are also discussed which ensure an operable and manageable system which functions uninterrupted and in a reliable fashion.

3. Detailed Design

3.1. Global system architecture

The system architecture is a 2-tier architecture which includes the stakeholder at one end and database at the other end. There is no server based middle tier used in the software.

3.2. Platform

Minimum system requirements:

Hardware Requirements:	
Operating system	Windows 98 or later versions, Linux
Processor	Pentium III processor or equivalent
Hard Disk space	500MB
RAM	512 MB

Recommended system requirements:

Hardware Requirements:	
Operating system	Windows 7 or later versions, Linux
Processor	Pentium IV processor or above
Hard Disk space	32 GB
RAM	1 GB

3.3. Software Architecture

Object-Oriented architecture is the basis of JIS. The data representations and their associated operations are encapsulated in an abstract data type or object. Objects interact through functions. The object is responsible for maintaining the integrity of its representation.

3.4 Report

In this section, the global system architecture and the minimum and required hardware and software requirements were mentioned. The software architecture of the software was also discussed.