SOFTWARE ENGINEERING

# Requirement Analysis & Understanding the sys. from an Analyst.

The person who is responsible to listen to the clients and documenting the under-standing so as the client & the develop-ment team would be able to work further on that project. This person is known as Analyst or System Analyst.

Other than collecting the requirements & documenting, the role of analyst also includes communicating properly with the team that will be responsible for initiating the development of the project.

The role of analyst also includes identifying & eliminating issues related to anomalies, inconsistency and incompleteness that may have arised during the interraction with the client or the development team members.

SRS → by Analyst → Sys. Req. Specification

This document prepared by the system analyst is the initial part of SRS (Software Requirement Specification). The imp parts of SRS documents are:

i) functional requirement of the system
   ↳ coding part

ii) Non-functional " " " "
   ↳ compatibility / characteristics

iii) Goals for implementation
   ↳ client's requirement

| Functional | Non-Functional | Goals |
|---|---|---|
| ↓ | ↓ | ↓ |
| abstract design | characteristics | client's req. |
| | ↓ | ↓ |
| | document | document |

## Properties of SRS document

i) <u>Concise</u>: The SRS must be consistent, complete and unambiguous. It shouldn't contain any irrelevant information that may lead to error.

ii) Well-Structured: The doc. must be well-structured such that it should be easy to understand & modify.

iii) Black-Box View: The doc. is not supposed to contain any tech. details (including designing, coding, testing or deployment details). Under this property, the SRS will describe external visible behaviour of the system.

iv) Conceptual integrity: The user can easily understand & implement the instructions written in SRS.

v) Verifyable: The information present in SRS must be correct & accepted.

vi) Response to undesired events: SRS must mention actions & response that need to be initiated when the sys. will face exceptional conditions.

# Decision Tree

A decision tree gives a graphic view of the processing logic involved in decision making & the corresponding action taken. The edge of a decision tree represents condition & the leaf nodes represents the actions. The actions are performed depending on the outcome of the testing cond$^{n}$. eg - you are supposed to develop a library automation system which should support the following:

i) new-member, ii) renew, iii) cancellation

→ Options for new member

Decision

When the new-member option is selected, the software asks the details about the member (name, add, contact no. etc).

Action

If proper information is entered, the membership record is created & the payment details (membership charge &

security deposit) are printed.

→ Renew membership

Decision

If the renew option is chosen then the software asks for member's name & membership no. to check the validity of the member.

Action

If the membership is valid then the expiry date is updated & annual membership charge is printed. Otherwise, an error message is displayed.

→ Cancel membership

Decision

If the cancel membership option is selected then the software ask for member's name & membership no. for processing.

Action

The membership is cancelled, membership record is deleted & membership fees is returned.