BCA - First Semester
ubject Name- “Software Engineering”
Paper-BCA-103

By- Prof. Dileep Kumar Sahu

Assistant Professor

Department of Computer Application

Govt. Vishwanath Yadav Tamaskar Post Graduate Autonomous College, Durg (C.G.)

Email ID: dileepksahu20@gmail.com

Unit III- Software Requirement and Specification

By- Prof DIleep Kumar Sahu, Assistant Professor, Govt. V.Y.T. PG
Auto. College Durg (C.G.)

ontent

1. Introduction and Need of SRS
2. Structured Analysis
   - Data Flow Diagram
   - Context Diagram
   - Data Dictionary

roduction to SRS

ftware Requirements Specification (**SRS)** is a document that lays out the cription of the software that is to be developed as well as the intention of software under development.

ftware requirements specification shows **what** the software is supposed do as well as **how** it is supposed to perform.

is written down before the actual software development work starts.

S is important for developers because it **minimizes** the amount of time d effort developers have to expend to achieve desired software goals.

thus **reduces** development cost. This also **benefits** the client company cause the lesser the development cost, the lesser the developers will arge from the client.

ntroduction to SRS

S should address, among other things:

ctionality of the software: What the software will do

rnal interfaces: How the given software will interact with
lware, other software and assumptions on these entities

uired performance levels: Required performance levels such as
onse rate, recovery rate etc. of the software

lity attributes: The non-functional factors that are used to
uate the performance of the software, such as security, safety,
ability etc

gn constraints: Any operating system limitations (e.g.: the stock
ange software will only run on Windows), implementation
uage etc that will affect or limit the design of the software.

ntroduction to SRS

nd, if composed properly, an **SRS** ensures that there is less possibility of ture redesigns as there is less chance of mistake on the part of developers they have a clear idea on the functionalities and externalities of the ftware.

also helps **clear** any communication problems between the **client** and e **developer**.

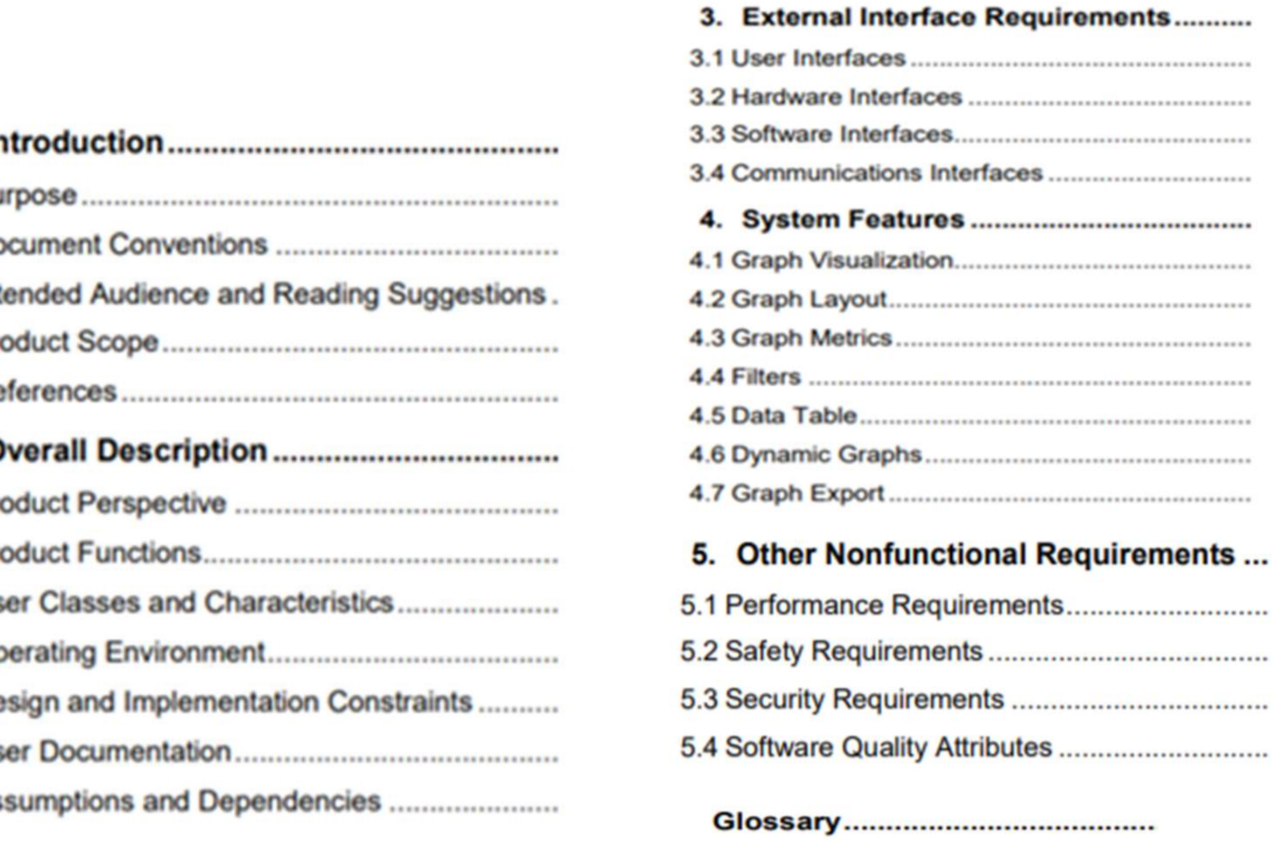
rthermore, an **SRS** serves to form a foundation of **mutual reement** between the client and the developer (supplier).

also serves as the document to **verify** the testing processes.

good SRS defines how an application will interact with system hardware, her programs and human users in a wide variety of real-world situations. rameters such as operating speed, response time, availability, portability, aintainability, security and speed of recovery from adverse events are aluated.

oduction to SRS

mat of software requirements specification given by **IEEE** (Institute of cal and Electronics Engineers) is shown below (explanation for each is given within the point):



rpose of an SRS

SRS forms the basis of an organization’s entire project.

ets out the framework that all the development teams will follow.

provides critical information to all the teams, including velopment, operations, quality assurance and maintenance, suring the teams are in agreement.

ng the SRS helps an enterprise confirm that the requirements are filled and helps business leaders make decisions about lifecycle of their product, such as when to retire a feature.

iting an SRS can help developers reduce the time and effort cessary to meet their goals as well as save money on the cost of velopment.

Structured Analysis

tructured Analysis is a development method that allows the analyst t
nderstand the system and its activities in a logical way.

t is a systematic approach, which uses graphical tools that analyze an
efine the objectives of an existing system and develop a new syste
pecification which can be easily understandable by user.

t has following attributes –

It is graphic which specifies the presentation of application.

It divides the processes so that it gives a clear picture of system flow.

It is logical rather than physical i.e., the elements of system do no
depend on vendor or hardware.

It is an approach that works from high-level overviews to lower-leve
details.

ctured Analysis Tools

ructured Analysis is diagrammatic notation which is design to help people derstand the system.

he basic goal of SA is to improve quality and reduce the risk of System ailure.

t establishes concrete management specification and documentation.

ing Structured Analysis, various tools and techniques are used for em development. They are −

a Flow Diagrams

a Dictionary

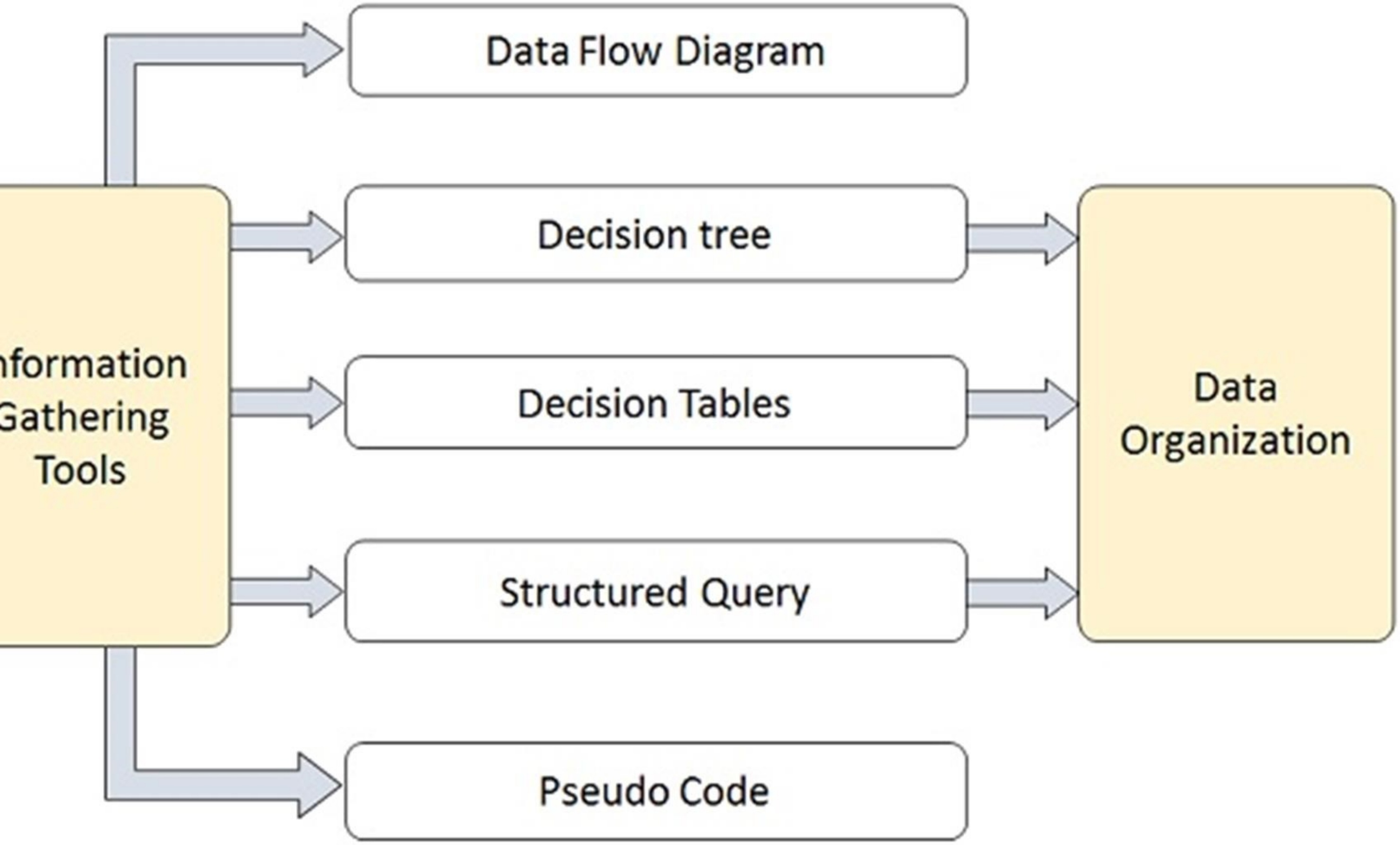
ision Trees

ision Tables

ctured English

udocode

uctured Analysis Tools



ta Flow Diagram

s a technique developed by Larry Constantine to express the uirements of system in a graphical form.

Data Flow Diagram (DFD) is a graphical representation of the ormation flows within a system.

DFD can depict the right amount of the system requirement phically. It can be manual, automated, or a combination of both.

hows flow of data enters and leaves the system, what types of nges the information, and where data is going to stored.

a Flow Diagram

objective of a DFD is to show the scope and boundaries of a em.

DFD may be used to perform a system or software at any level of traction.

nay be used as a communication tool between a system analyst any person who plays a part in the order that acts as a starting nt for redesigning a system.

e DFD is also called as a data flow graph or bubble chart.

By- Prof DIleep Kumar Sahu, Assistant Professor, Govt. V.Y.T. PG
Auto. College Durg (C.G.)

ic Elements of Data Flow Diagram

ymbol	Symbol Name	Function
→	Arrow	Data Flow , used to connect the process
□	Box or Rectangle	Source of System Input and Destination of System Output
○	Circle	It is a Process, it transforming data flow
=	Open Rectangle	Data Store

By- Prof DIleep Kumar Sahu, Assistant Professor, Govt. V.Y.T. PG
Auto. College Durg (C.G.)

a Flow Diagram

te:

l names should be unique.

emember that DFD is not a flow chart. arrows in DFD represents owing data.

DFD does not involve any order of events.

o not become bogged down with details. Defer error conditions and rror handling until the end of the analysis.

es of Data Flow Diagram

s are of two types: Physical DFD and Logical DFD.

ollowing table lists the points that differentiate a physical DFD from a logical DFD.

Physical DFD	Logical DFD
nplementation dependent. It shows functions are performed.	It is implementation independent. It focuses only on the flow of data between processes.
vides low level details of hardware, are, files, and people.	It explains events of systems and data required by each event.
icts how the current system operates ow a system will be implemented.	It shows how business operates; not how the system can be implemented.

els in Data Flow Diagram

Ds may be partitioned into levels that represent increasing ormation flow and functional detail.

vels in DFD are numbered 0, 1, 2 or beyond.

re, we will see primarily three levels in the data flow diagram, ich are: 0-level DFD, 1-level DFD, and 2-level DFD.

evels Data Flow Diagram

is also called context diagram.

represents the entire software requirement as a single bubble with put and output data denoted by incoming and outgoing arrows.

bubble "A" has two inputs x_1 and x_2 and one output y, then the panded DFD, that represents "A" should have exactly two external puts and one external output as shown in fig:

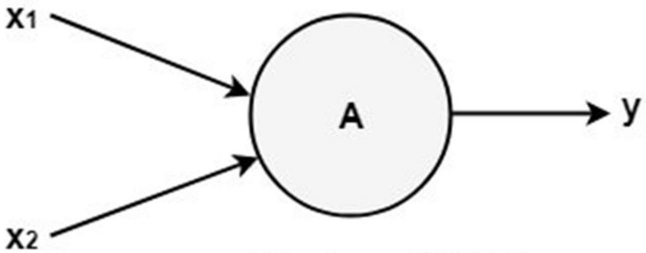


Fig: Level-0 DFD.

evels Data Flow Diagram- Example

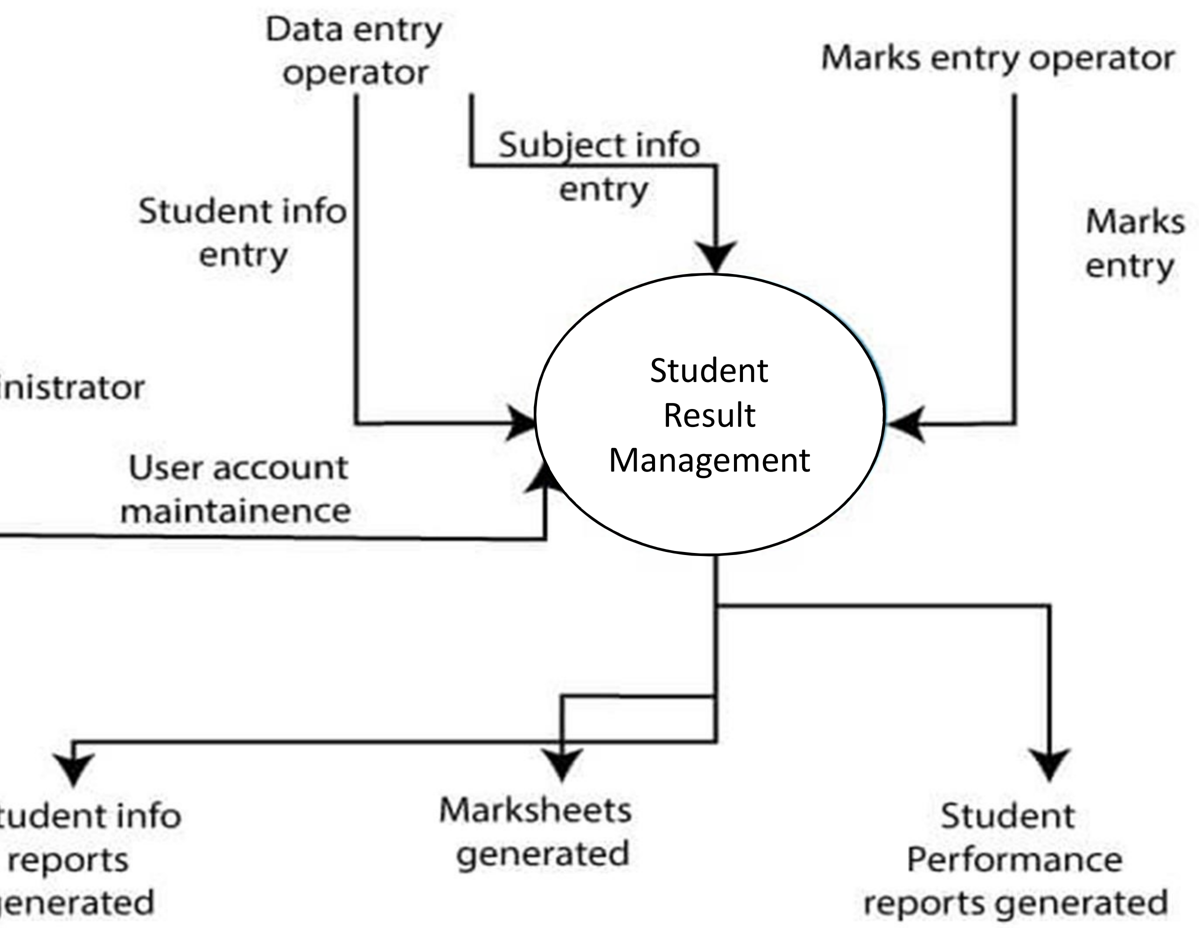


Fig: Level-0 DFD of result management system

By- Prof DIleep Kumar Sahu, Assistant Professor, Govt. V.Y.T. PG Auto. College Durg (C.G.)

vels Data Flow Diagram

DFD, O-level DFD is decomposed into multiple processes. In this level, we highlight the main ves of the system and breakdown the high-level process of 0-level DFD into sub-processes.

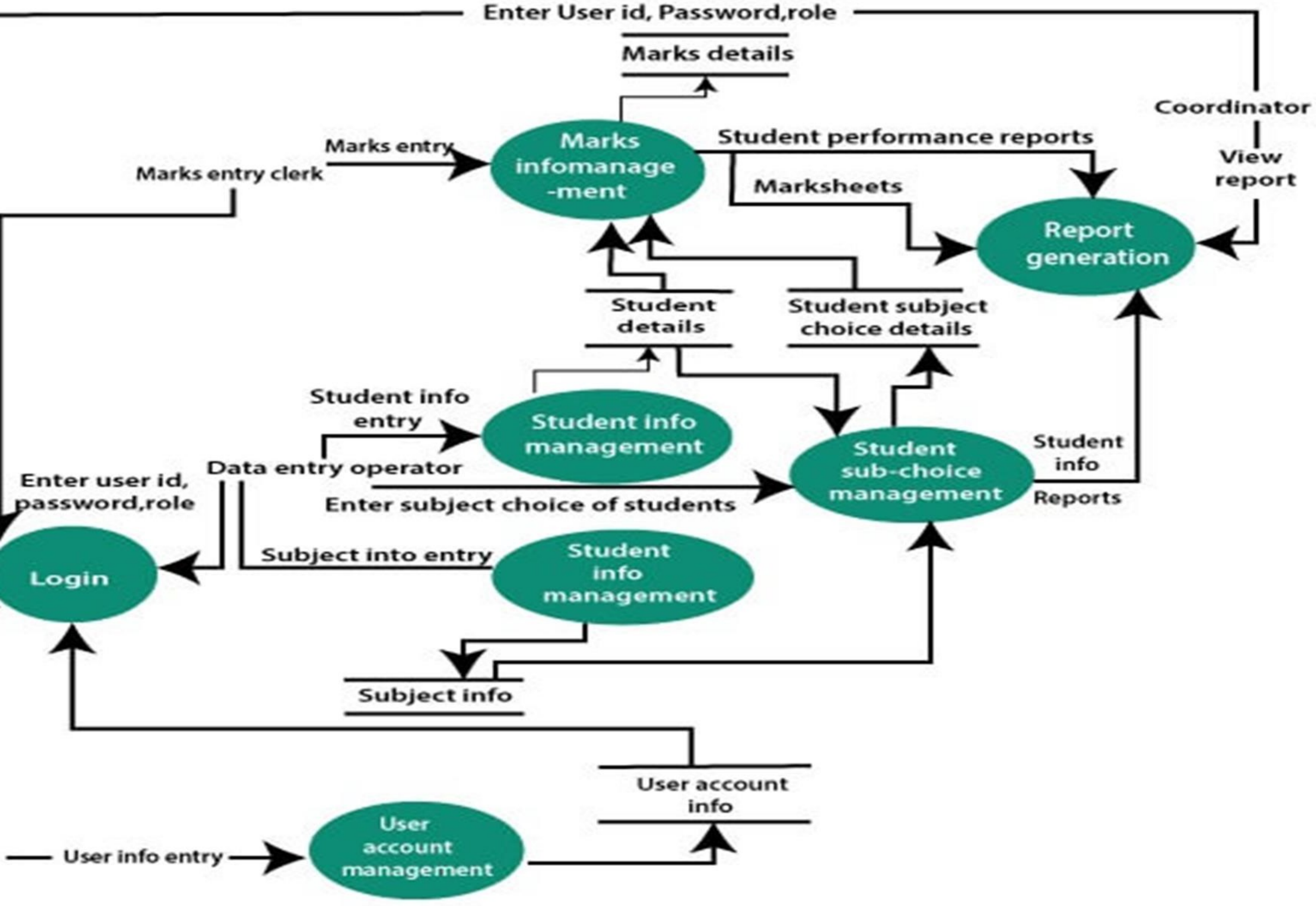


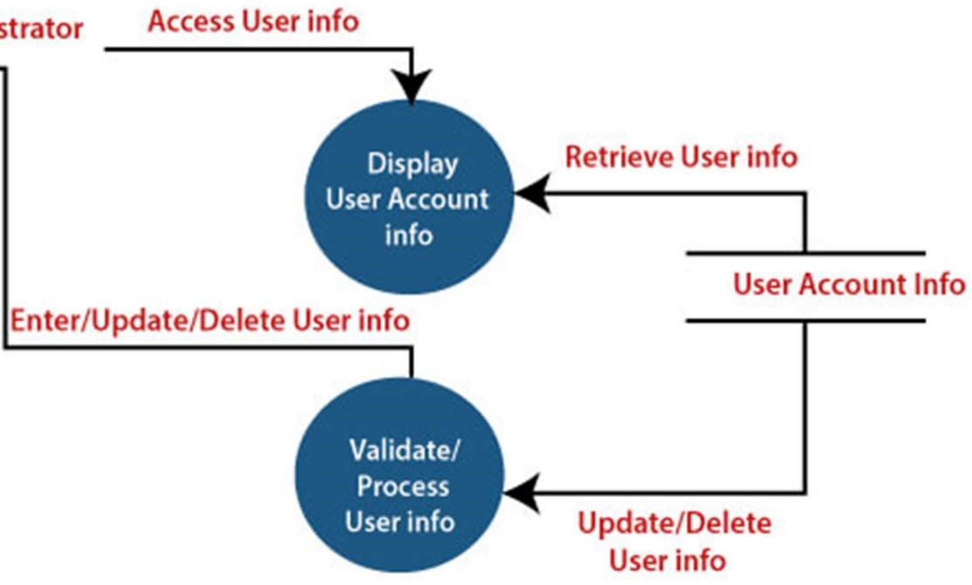
Fig: Level-1 DFD of result management system

evels in Data Flow Diagram

el DFD goes one process deeper into parts of 1-level DFD.

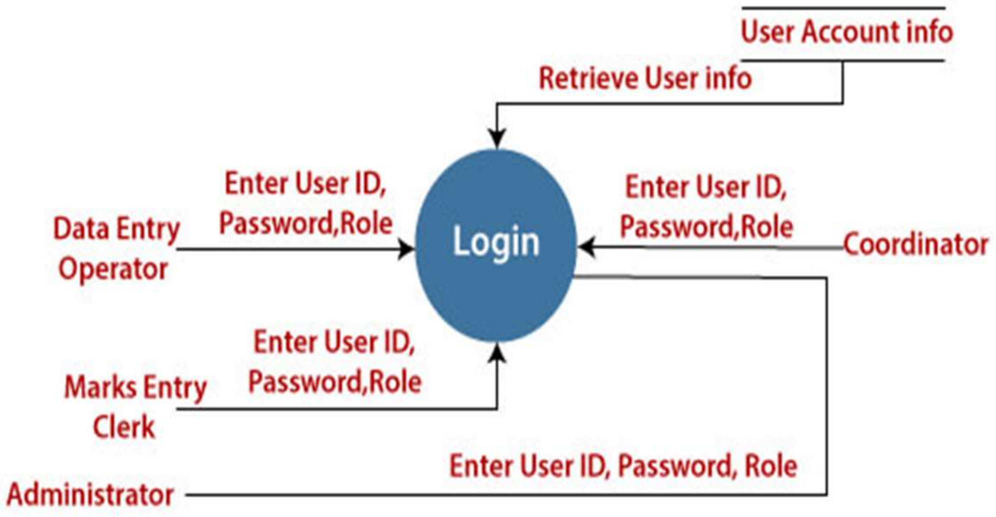
n be used to project or record the specific/necessary l about the system's functioning.

Account Maintenance


strator
Access User info
Display User Account info
Retrieve User info
User Account Info
Enter/Update/Delete User info
Validate/ Process User info
Update/Delete User info

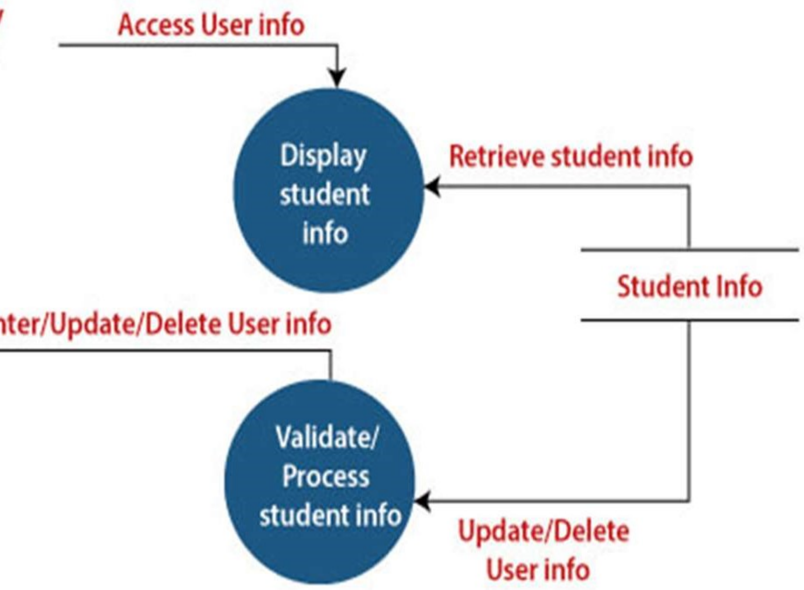
2. Login

The level 2 DFD of this process is given below:


User Account info
Retrieve User info
Enter User ID, Password,Role
Data Entry Operator
Login
Enter User ID, Password,Role
Coordinator
Enter User ID, Password,Role
Marks Entry Clerk
Enter User ID, Password, Role
Administrator

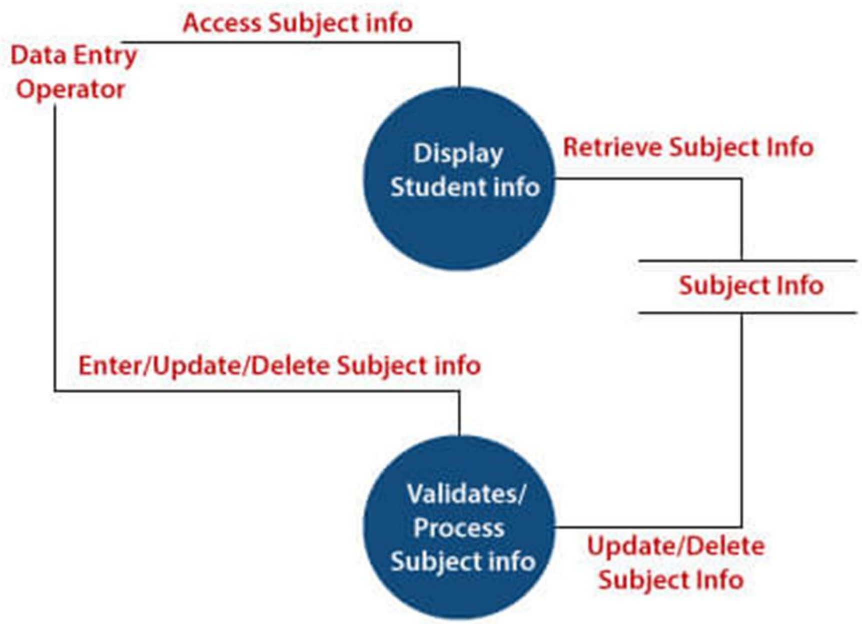
evels in Data Flow Diagram

nt Information Management



4. Subject Information Management

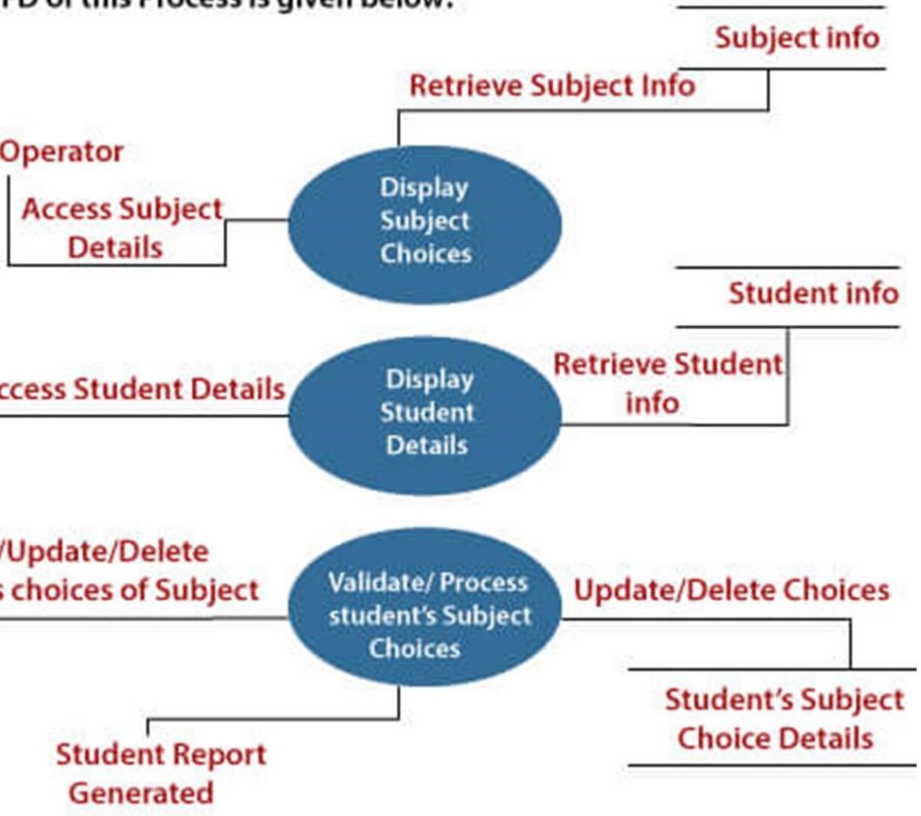
The level 2 DFD of this process is given below:



evels in Data Flow Diagram

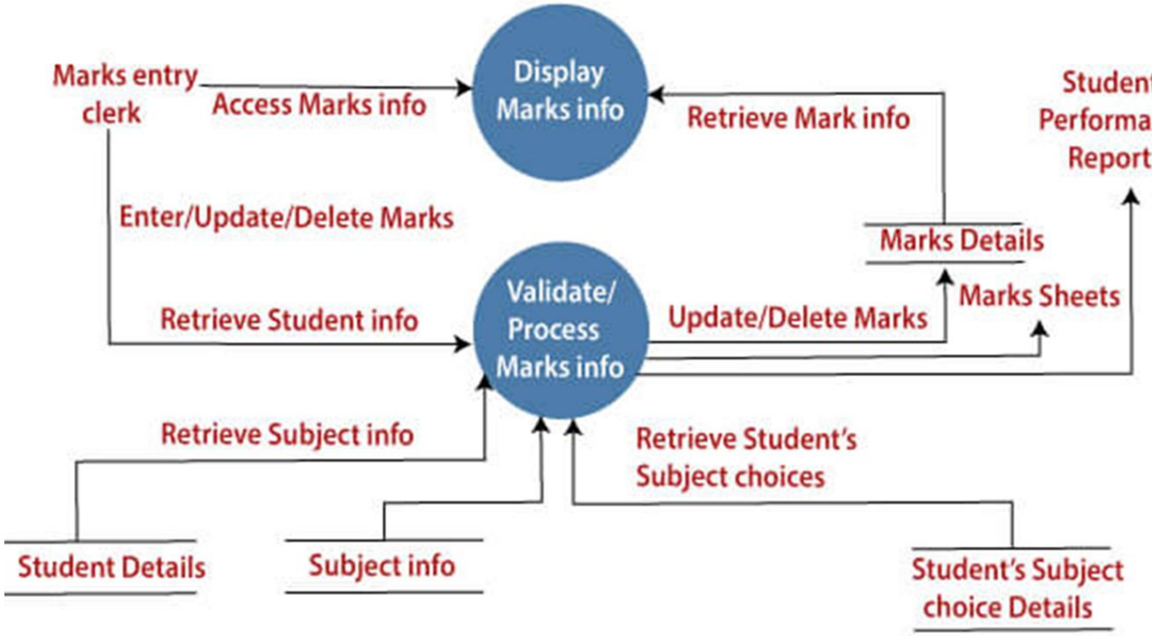
s Subject Choice Management

FD of this Process is given below:



6. Marks Information Managment

The Level 2 DFD of this Process is given below:



a Dictionary

ta dictionary is a file or a set of files that includes a database's metadata.

data dictionary hold records about other objects in the database, such as ownership, data relationships to other objects, and other data.

data dictionary is an essential component of any relational database.

ause of its importance, it is invisible to most database users.

cally, only database administrators interact with the data dictionary.

ta dictionary improves the communication between the analyst and the

.

ays an important role in building a database.

a Dictionary

lata dictionary includes information about the following:

ne of the data item

ses

cription/purpose

ated data items

ge of values

a structure definition/Forms

a Dictionary

he **name of the data item** is self-explanatory.

iases: include other names by which this data item is called EO for Data Entry Operator and DR for Deputy Registrar.

escription: is a textual description of what the data item is ed for or why it exists.

elated data items: capture relationships between data items g., total_marks must always equal to internal_marks plus ternal_marks.

a Dictionary

eld Length: records all possible values, e.g. total marks must e positive and between 0 to 100.

ata structure Forms: Data flows capture the name of ocesses that generate or receive the data items.

If the data item is primitive, then data structure form captures the physical structures of the data item.

If the data is itself a data aggregate, then data structure form capture the composition of the data items in terms of other data items.

a Dictionary: Student Table

ield Name	Data type	Field Length	Description
oll_no	Int	10	Primary Ke, Auto Generated
tud_name	Varchar	20	Student name
tud_class	Varchar	15	Class of student
mailid	Varchar	15	Email id of student
ontact	In	15	Contact number of student

By- Prof DIleep Kumar Sahu, Assistant Professor, Govt. V.Y.T. PG
Auto. College Durg (C.G.)

lications of Data Dictionary

- *Create an ordered listing of all data items.*
- *Create an ordered listing of subset of data items.*
- *Find a data item name from a description.*
- *Design the software and test cases.*

antages of Data Dictionary

/ using a data dictionary, designers have a central point of ntrol and documentation for making changes and updates to ita sets and structures.

also helps programmers working on updating or replacing gacy systems figure out the basic data structures of the stems they are renovating and the definitions of individual riables.

lso, it allows easy development of new or ad hoc queries sing SQL.

Thank You