KYAMBOGO UNIVERSITY

SCHOOL OF COMPUTING AND INFORMATION SCIENCE.

DEPARTMENT OF COMPUTER SCIENCE.

BACHELOR OF INFORMATION SYSTEMS -(BIS).

COURSE UNIT: **INFORMATION SYSTEMS ANALYSIS AND DESIGN.**
LECTURER: **MRS. NANTONGO BARTHA ALEXANDAR.**
TASK: **INDIVIDUAL COURSEWORK.**
YEAR: **TWO**
SEMESTER: **ONE**

STUDENT'S DETAILS

NAME	*REG. NUMBER*	*SIGNATURE*
LUTAAYA JAMIL	20/U/ISD/9608/PD	

Question:

a) Discuss the importance of Data Flow Diagram (DFD) in system analysis and design.
b) Using an example, explain and illustrate;
   i. DFD level 0.
   ii. DFD level 1.

A **data flow diagram** is a graphical portrayal of the flow of information in a system.

In other words, a **data flow diagramming** is a technique that diagrams the business processes and the data that pass among them. Data flow diagrams focus specifically on the processes and the data that passes through them, not strictly just data. (*Wang & Wang,* 2001).

There are two types of data flow diagrams that is ***logical*** and ***physical flow diagrams***.

A ***logical flow diagram*** is one which focuses on activities of the business and business as a whole. It is the basis of the physical flow diagram. They explain the business events and data needed to accomplish them.

A ***physical flow diagram*** is developed from a logical flow diagram. These provide detailed information about the hardware, software and other external components.

Elements of data flow diagrams.

There are four symbols in the Data Flow Diagram (DFD) language, that is;

- Processes
- Data flows
- Data stores
- External entities

They are respectively represented by individual graphical symbols. There are two commonly used symbols' styles, one was developed by Chris Gane and Trish Sarson and the other by Tom DeMarco and Ed Yourdon. Though other sources may be using more.

i. Processes:

A process is an operation that alters data and produces an output. It may be carrying out calculations, sort data according to logic, and many other actions.

Processes can be manual or computerized.

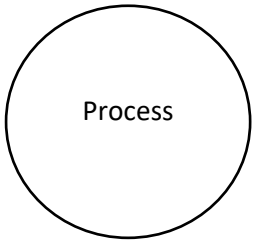

Figure 1: DeMarco and Yourdon symbol of a process.

ii. External entity:

An external entity is a person, organization unit or system that is external to the system but communicates with the diagrammed system by sending or receiving data.

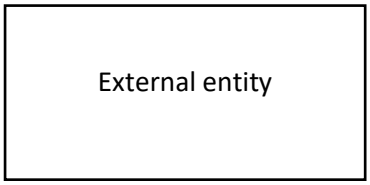

Figure 2: DeMarco and Yourdon symbol of an external entity.

iii. Data flow:

Data flow is the route taken buy data between external entities, processes and data stores. This acts as an interface between these components.

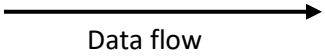

Figure 3: Symbol of a data flow.

iv. Data store:

A data store is where data is stored and retrieved in the system. This can be in any form, either a file, book or something else. *Leclerc, J., & Desbiens, D.* (2011).

Figure 4: DeMarco and Yourdon symbol of a data store.

Rules and Guidelines for DFDs

- There should be at least one input and one output for each procedure.
- Each data storage should have at least one data flow in, and one data flow out.
- Names should be unique. This makes referring to elements in the DFD easy.
- To save data in a system, a process must be followed.
- Processes should lead to other processes or saving something.
- Data stored in a system must go through a process.

a. Importance of Data Flow Diagrams.

- They enable us understand how a system operates and thus being able to identify limitations.
- It aids in visualizing the contexts of a system, here even a person with low or no technical or programming skills can be able to understand what is being done.
- It is included in the system's documentation file which provides additional assistance in terms of reference to other system users who are implementing or maybe adjusting.
- Because of using arrows and other symbols, Data Flow Diagrams makes the system easy to explain to other parties which eases collaboration.*(Valacich et al.* 2008*).*
- They enable analyst to critically observe data and so can capture redundant data. This also enables analysts to view data that gives no output in the system.
- It makes presentation of the system easy, since different people have to know about what the system is going to do and how it is going to tackle them, programmers cannot use codes to illustrate, so the DFDs ease this task.

Though Data Flow Diagrams are good, they also have some few shortcomings, they are below;

- They relatively take a longer time to design which at most time makes many organizations to deny analysts from making them.
- They also make programmers who are to implement the system confused. Since most of them tend to be complex.

b. There are different levels in Data Flow Diagrams' concept starting from 0, 1, 2 and sometimes beyond. Some scholars differentiate the context diagram and level 0, though in this document am going to use it as same level according to majority.

i. **DFD level 0**.

Level 0 is also called the "context diagram". It is the basic overview of the whole system or process being studied or focused on. It gives us a glimpse of the system as a simple process that is connected to external entities.

I am to use a course enrollment system as an example.

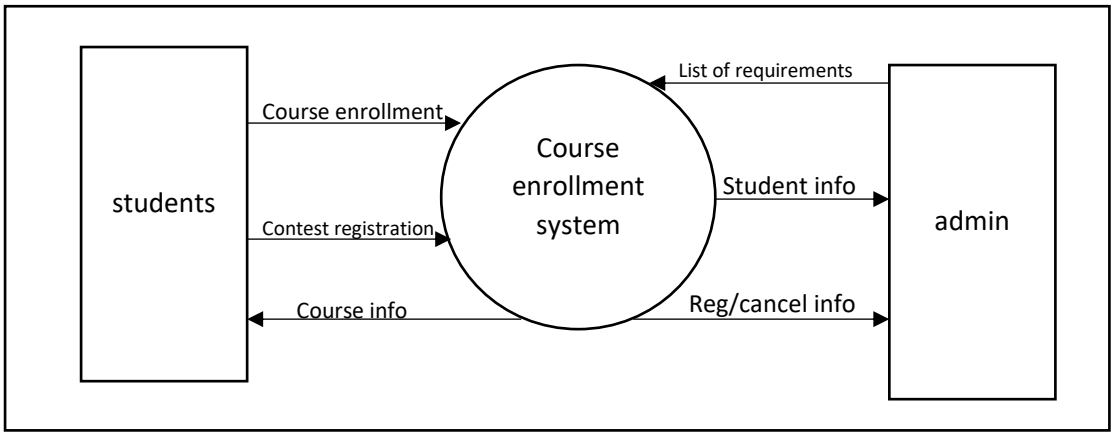

Figure 5: DFD level 0.

The course enrollment system portrayed at level 0 shows how a student goes through the enrollment process. He has to go through the 'course enrollment', hence sending request to the system which further contacts the administrator by provided him/her with the student information who gives feedback with the list of requirements for one to enroll which is processed and sent to the student in "course info".

ii. DFD level 1.

The DFD level 1 gives us a more detailed breakdown of level 0's components. It gives us a clearer look into the sub components of the system. *(Data Flow Diagrams in System Analysis - Coding Ninjas CodeStudio, 2022).*

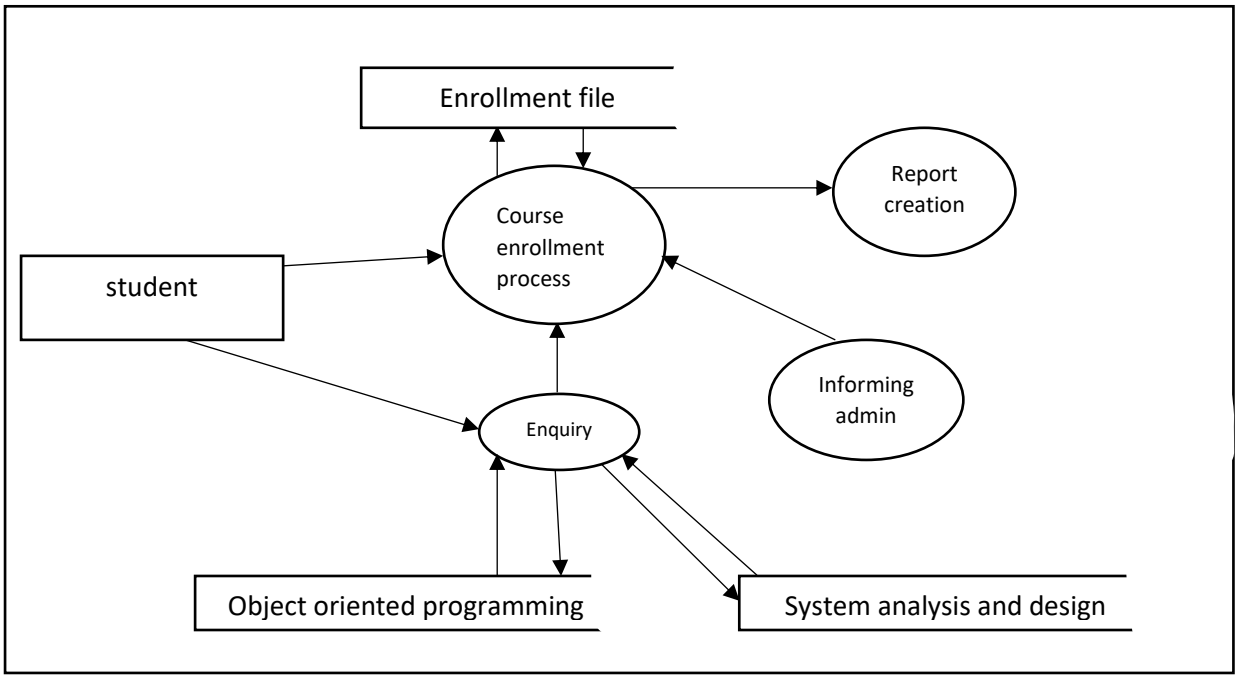

Figure 6: DFD level 1.

The diagram in *figure 6* above shows DFD level 1. An example course enrollment process is used (expanded) from the context diagram.

It shows that a student sends a request to the *course enrollment process* which checks an *enrollment file* for the student details. Feedback is received from the file to course enrollment process. The enrollment process sends information to the *report creation* process which also gives input to another process and data store then it returns feedback to the course enrollment process, the line of feedback from *report creation* is not shown because the report process has further actions to perform which are not shown. More processes can be expanded in level one either on one page o different pages according to the designer.

References

Wang, S., & Wang, H. (2001). *INFORMATION SYSTEMS ANALYSIS AND DESIGN.*

Leclerc, J., & Desbiens, D. (2011). *SYSTEMS ANALYSIS & DESIGN An Object-Oriented Approach.* (5th ed.). Ottawa: Douglas & McIntyre.

Valacich, J. S., George, J. F., & Hoffer, J. A. (2008). *Global edition essentials of Systems analysis and design Sixth edition.*

Data Flow Diagrams in System Analysis - Coding Ninjas CodeStudio. (2022). Retrieved June 26, 2022, from https://www.codingninjas.com/codestudio/library/data-flow-diagrams-in-system-analysis.