**SYSTEM DESIGN**

Designing is the most important phase of software development. It requires a careful planning and thinking on the part of the system designer. Designing software means to plan how the various parts of the software are going to achieve the desired goal. It should be done with utmost care because if the phase contains any error then that will affect the performance of the system, as a result it may take more processing time, more response time, extra coding workload etc.

Software design sits at the technical kernel of the software engineering process and is applied regardless of the software process model that is used. After the software requirements have been analyzed and specified, software design is the first of the three technical activities Designing, Coding and Testing that are required to build and verify the software. Each activity transforms information in such a manner that ultimately results in validated computer software.

**DESIGN GOALS**

**T**he following goals were kept in mind while designing the system:

* ***Make system user-friendly****.* This was necessary so that system could be used efficiently and system could act as catalyst in achieving objectives.
* ***Make system compatible*** *i.e.* It should fit in the total integrated system. Future maintenance and enhancement must be less.
* **Make the system compatible** so that it could integrate other modules of system into itself.
* **Make the system reliable, understandable and cost-effective**.

It is the most critical factor affecting the quality of the software.

1. Logical Design
2. Physical Design
3. Input Design
4. Output Design
5. User Interface Design
6. Code Design.
7. Database Design

**LOGICAL DESIGN**

The logical design seeks to trace the flow of data throughout the system. Data flow diagram where designed to achieve this end in a graphical format that is easy to understand. All process on any of the DFDs must have at least one input and output dataflow.

**PHYSICAL DESIGN**

The physical design takes into consideration the physical data flows, which must represent any of the following.

* The planned implementation of an input to or output from a physical process.
* A database command or action such as insert, delete, and update.
* The import of data from or export of data to another information system across a network.
* The flow of data between to modules or subroutines within the same program.

**INPUT DESIGN**

Inputs can be classified according to two characteristics.

* How data is initially captured, entered and processed.
* The method and technology used to capture and enter data.
* Input begins long before the data arrives at the input Device, be it a keyboard or a mouse. Source documentation, input screens methods and procedures for getting the data into the computer have to be designed first.

**OUTPUT DESIGN**

Design of output screens has been kept as simple as possible. The user is provided with either a tabular representation or a statement giving details of transaction.

**USER INTERFACE DESIGN**

The basic steps or user interface design have been followed. They are

* Managing the user interface dialogue.
* Prototyping the dialogue and user interface.
* Obtaining user feedback.

**CODE DESIGN**

The main purpose of code design is to facilitate the identification and retrieval of information. Code design is the process of representing data flow diagram. This should be easy to debug when the error occurs.

**DATABASE DESIGN**

The data pertaining to proposed system is voluminous that a careful design of the database must proceed before storing the data in the database.

A database management system provides flexibility in the storage and retrieval of data bad production of information. The DBMS is a bridge between the application program, which determines what data are needed and how they are processed, and the operating system of the computer, which is responsible for placing data on the magnetic storage devices. A schema defines the database and a subschema defines the portion of the database that a specific program will use.

**Data Model**

The organization of the data is represented by a data model and identifies the logical organization of data. In a model of real world similar things are usually grouped into classes of object called object types.

A data model is a pattern according to which data are logically organized. It consists of the named logical units of data and expresses the relationships among the data as determined by the interpretation of the model of real world.

The relational data model is a formal model for representing relationships among attributes of an entity set and the association between entity sets.

In the relational data model all attribute relationships and all associations are represented as relations. There is no distinction even at the model level, between the different kinds or relations. Syntactically all the relations are the same. The data model does not produce the introduction of additional semantic information to distinguish different relations according to their properties of models namely:

* Network Model
* Hierarchical data model
* Relational Data model

**Relational data Model**

The relational data model is a formal model for representing relationships among attributes of an entity set and the association between entity sets.

In the relational data model all attribute relationships and all associations are represented as relations. There is no distinction even at the model level, between the different kinds or relations. Syntactically all the relations are the same. The data model does not produce the introduction of additional semantic information to distinguish different relations according to their properties.