**Godavari College Of Engineering, Jalgaon.**

**Subject Name:** PDE **Teacher Name:** Prof.Nilesh Chaudhari Sir.

**Practical No**. : 4 **Date:**

**Class: S**.E **Roll No: 1951711245011**

**Title:**  Explain DFD ( Data Flow Diagram.) in Software Engineering.

**Data Flow Diagram. :-** A data flow diagram, or a DFD, is a visual representation of any process or system’s flow of information. By mapping out your process or system’s flow of data, DFDs help you better understand your process or system, uncover its kinks, improve it, and can even help you implement a new process or system. DFDs can range from simple overviews to complex, granular displays of a process or system.

DFDs became popular in the 1970s and have been able to maintain their widespread use by being easy to understand. Visually displaying how a process or system works can hold people’s attention and explain complex concepts better than blocks of text can, so DFDs are able to help almost anyone grasp a system or process’ logic and functions.

**There are two types of DFDs** -- logical and physical. Logical diagrams display the theoretical process of moving information through a system, like where the data comes from, where it goes, how it changes, and where it ends up.

Physical diagrams shows you the practical process of moving information through a system, like how your system’s specific software, hardware, files, employees, and customers influences its flow of information.

You can either use logical or physical diagrams to describe the same flow of information or you can use them in conjunction to understand a process or system on a more granular level. But before you can use a DFD to understand your system or process’ flow of information, you need to know the standard notations or symbols used to describe it.

a) **Logical data flow diagrams** focus on what happens in a particular information flow: what information is being transmitted, what entities are receiving that info, what general processes occur, etc. The processes described in a logical DFD are business activities—a logical DFD doesn’t delve into the technical aspects of a process or system. Non-technical employees should be able to understand these diagrams.

b) **Physical data flow diagrams** focus on how things happen in an information flow. These diagrams specify the software, hardware, files, and people involved in an information flow. A detailed physical data flow diagram can facilitate the development of the code needed to implement a data system.

## **DFD components**

DFD consists of processes, flows, warehouses, and terminators. There are several ways to view these DFD components.

**Process**

The process (function, transformation) is part of a system that transforms inputs to outputs. The symbol of a process is a circle, an oval, a rectangle or a rectangle with rounded corners (according to the type of notation). The process is named in one word, a short sentence, or a phrase that is clearly to express its essence.

**Data Flow**

Data flow (flow, dataflow) shows the transfer of information (sometimes also material) from one part of the system to another. The symbol of the flow is the arrow. The flow should have a name that determines what information (or what material) is being moved. Exceptions are flows where it is clear what information is transferred through the entities that are linked to these flows. Material shifts are modeled in systems that are not merely informative. Flow should only transmit one type of information (material).

**Warehouse**

The warehouse (datastore, data store, file, database) is used to store data for later use. The symbol of the store is two horizontal lines, the other way of view is shown in the DFD Notation. The name of the warehouse is a plural noun (e.g. orders) - it derives from the input and output streams of the warehouse. The warehouse does not have to be just a data file, for example, a folder with documents, a filing cabinet, and optical discs. Therefore, viewing the warehouse in DFD is independent of implementation. The flow from the warehouse usually represents the reading of the data stored in the warehouse, and the flow to the warehouse usually expresses data entry or updating (sometimes also deleting data).

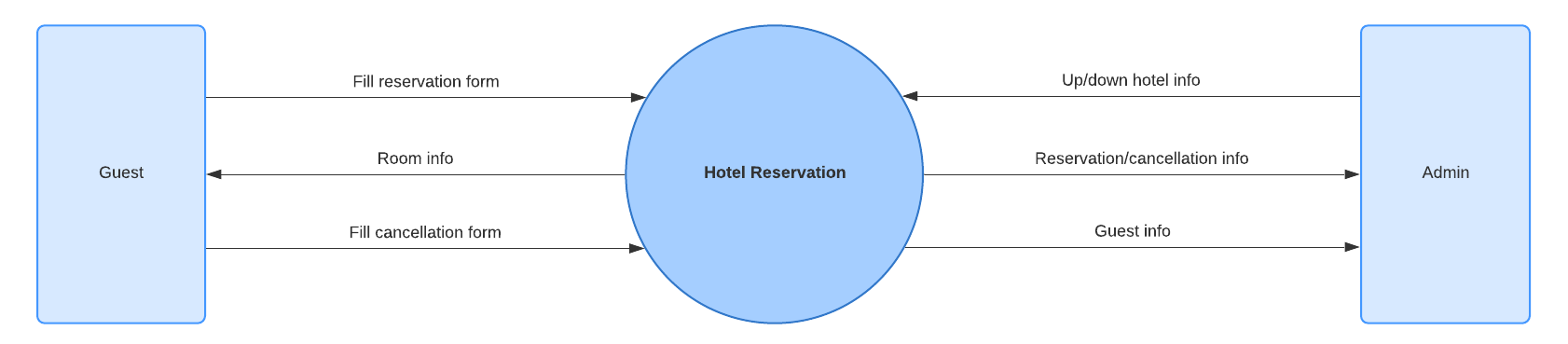
**Terminator**

The Terminator is an external entity that communicates with the system and stands outside of the system. It can be, for example, various organizations (eg a bank), groups of people (e.g. customers), authorities (e.g. a tax office) or a department (e.g. a human-resources department) of the same organization, which does not belong to the model system. The terminator may be another system with which the modeled system communicates.

## **Data flow diagram levels**

Data flow diagrams are also categorized by level. Starting with the most basic, level 0, DFDs get increasingly complex as the level increases. As you build your own data flow diagram, you will need to decide which level your diagram will be.

Level 0 DFDs, also known as context diagrams, are the most basic data flow diagrams. They provide a broad view that is easily digestible but offers little detail. Level 0 data flow diagrams show a single process node and its connections to external entities.

 Fig. Data Flow Diagram Level 0 Template

Level 1 DFDsare still a general overview, but they go into more detail than a context diagram. In a level 1 data flow diagram, the single process node from the context diagram is broken down into subprocesses. As these processes are added, the diagram will need additional data flows and data stores to link them together.

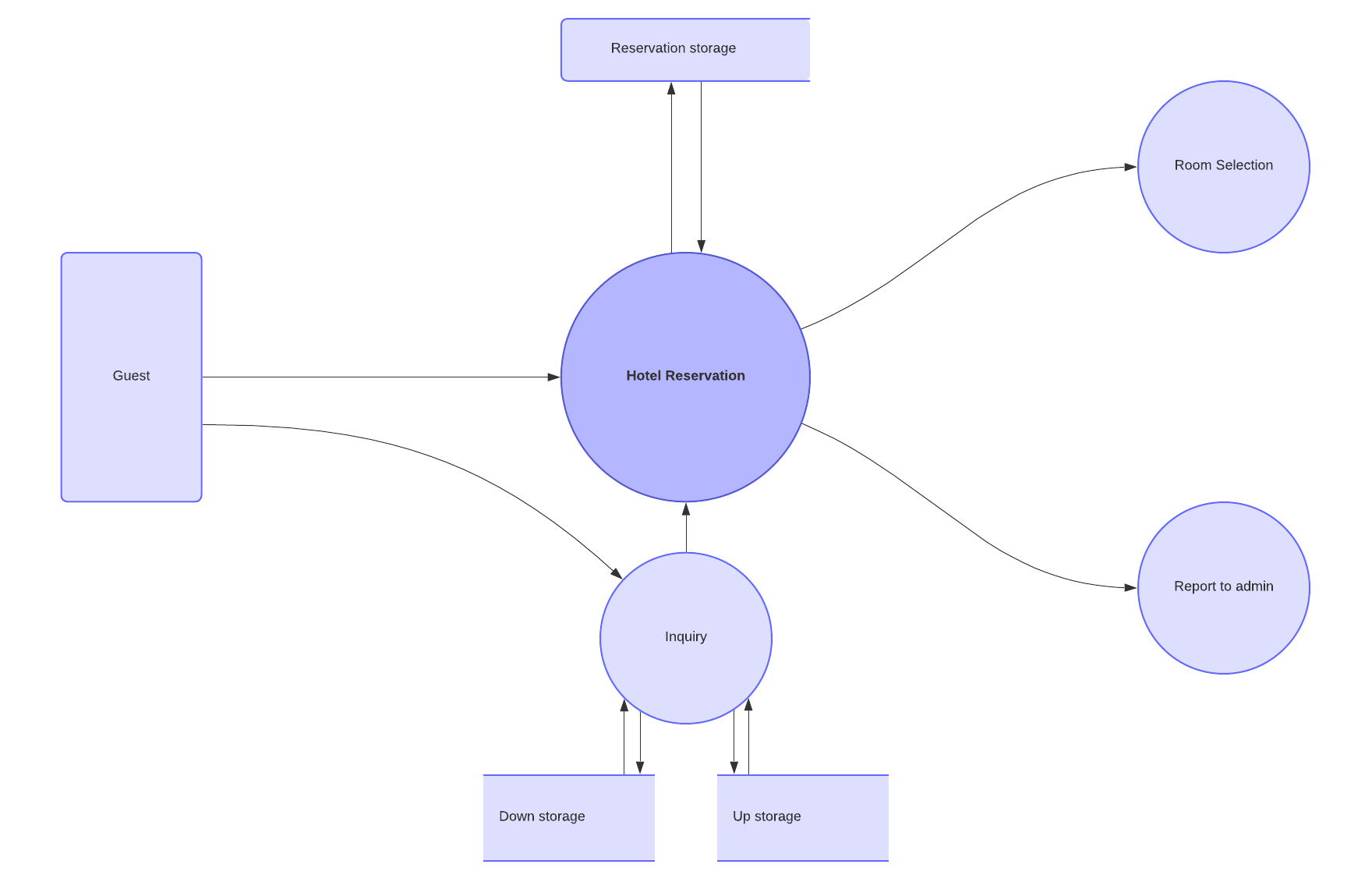


Fig.Data Flow Diagram Level 1 Template

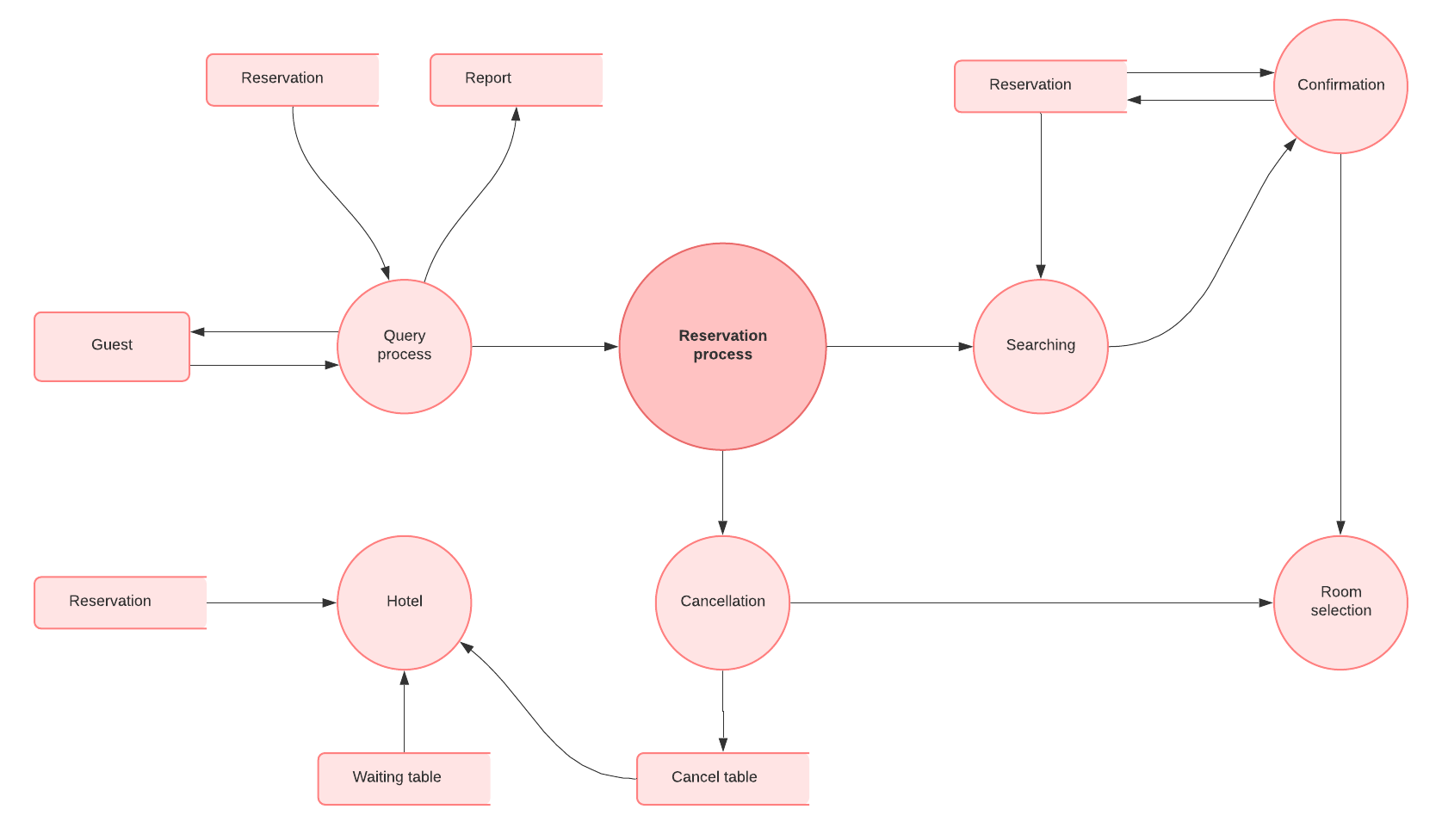
Level 2+ DFDs simply break processes down into more detailed subprocesses. In theory, DFDs could go beyond level 3, but they rarely do. Level 3 data flow diagrams are detailed enough that it doesn’t usually make sense to break them down further.

Fig. Data Flow Diagram Level 2 Template

**Conclusion:-** I learn from this Assignment what is DFD (Data Flow Diagram. ) and how to implemnt it in Software Engineering.