
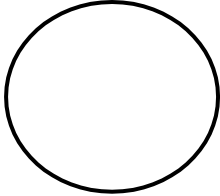
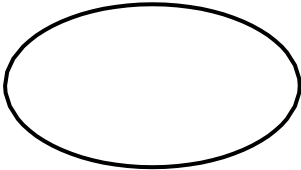




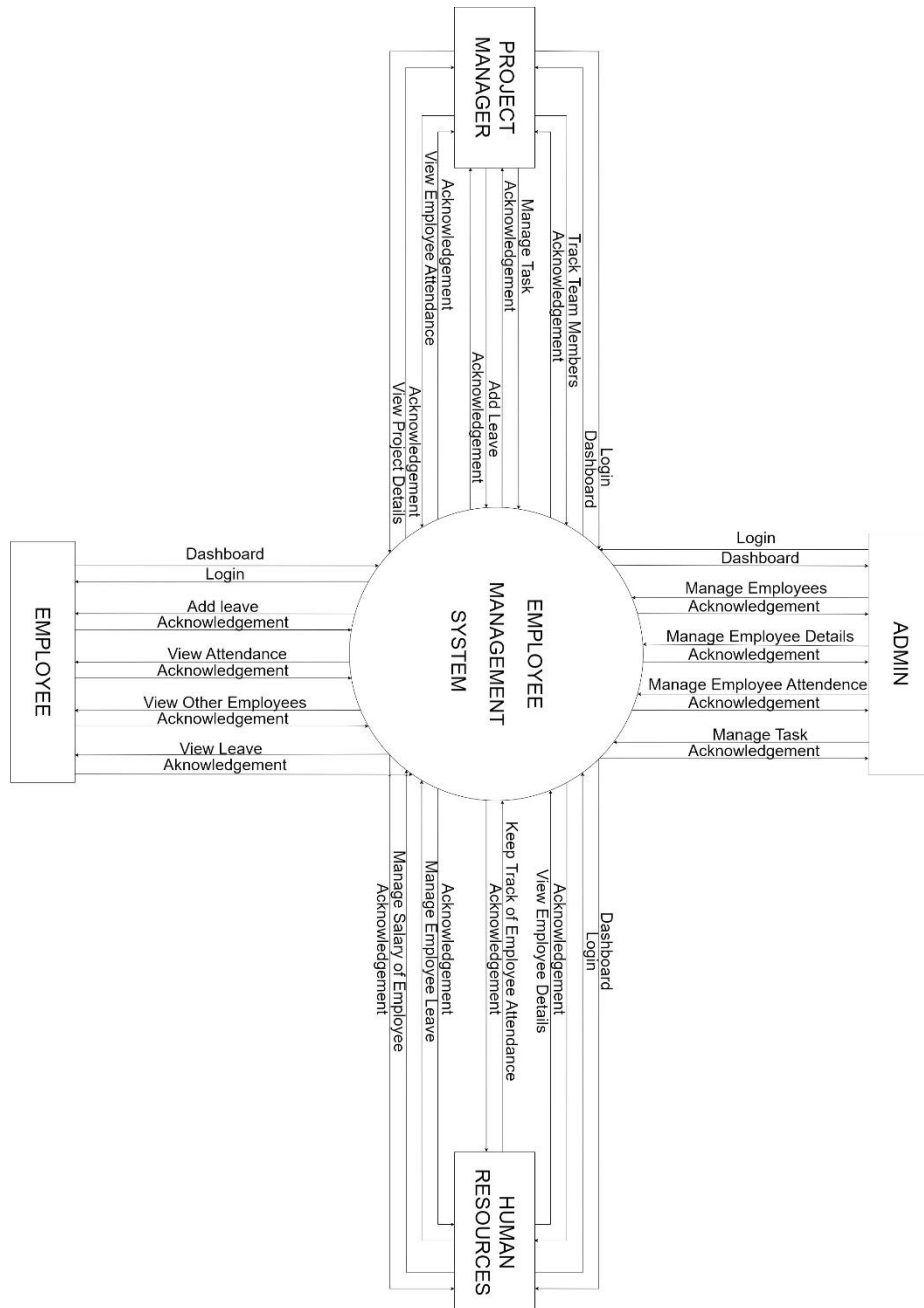
3.1 Data Flow Diagram

- DFD (data flow diagram) is also known as bubble chart or data flow graph.
- DFD's are very useful in understanding the system and can be effectively used during analysis. It shows flow of data through a system visually. The DFD is a hierarchical graphical model of a system the different processing activities or functions that the system performs and the data interchange among these functions.
- It views a system as a function that transforms the inputs into desired output.
- Each function is considered as a process that consumes some input data and produces some output data.
- Function model can be represented using DFD.
- DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system.
- The visual representation makes it a good communication tool between User and System designer.
- Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams.
- DFD has often been used due to the following reasons:
 1. Logical information flow of the system.
 2. Determination of physical system construction requirements.
 3. Simplicity of notation.
 4. Establishment of manual and automated systems requirements.

[Table 2: Data Flow Diagram Symbols]

Symbols	Description
	Entity: Entities are external to the system which interacts by inputting the data.
	System: It shows the system name.
	Process: It shows the part of the system that transforms into outputs.
	Data Flow: It passes the data from one part to another.
	Data Store: Data store is represented by two parallel lines. It is generally logical file or database.

Level 0: CONTEXT



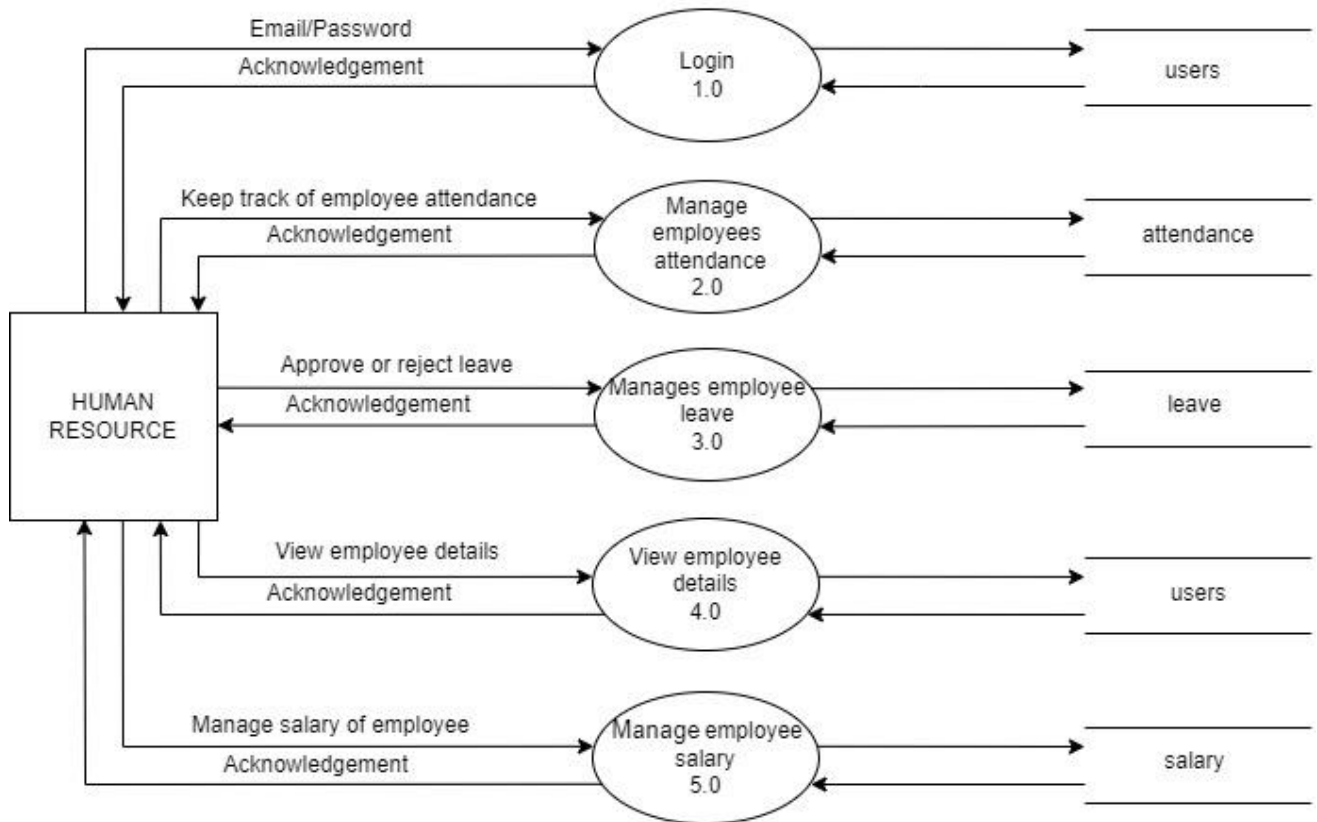
[Figure 2: Context level]

Level 1: ADMIN



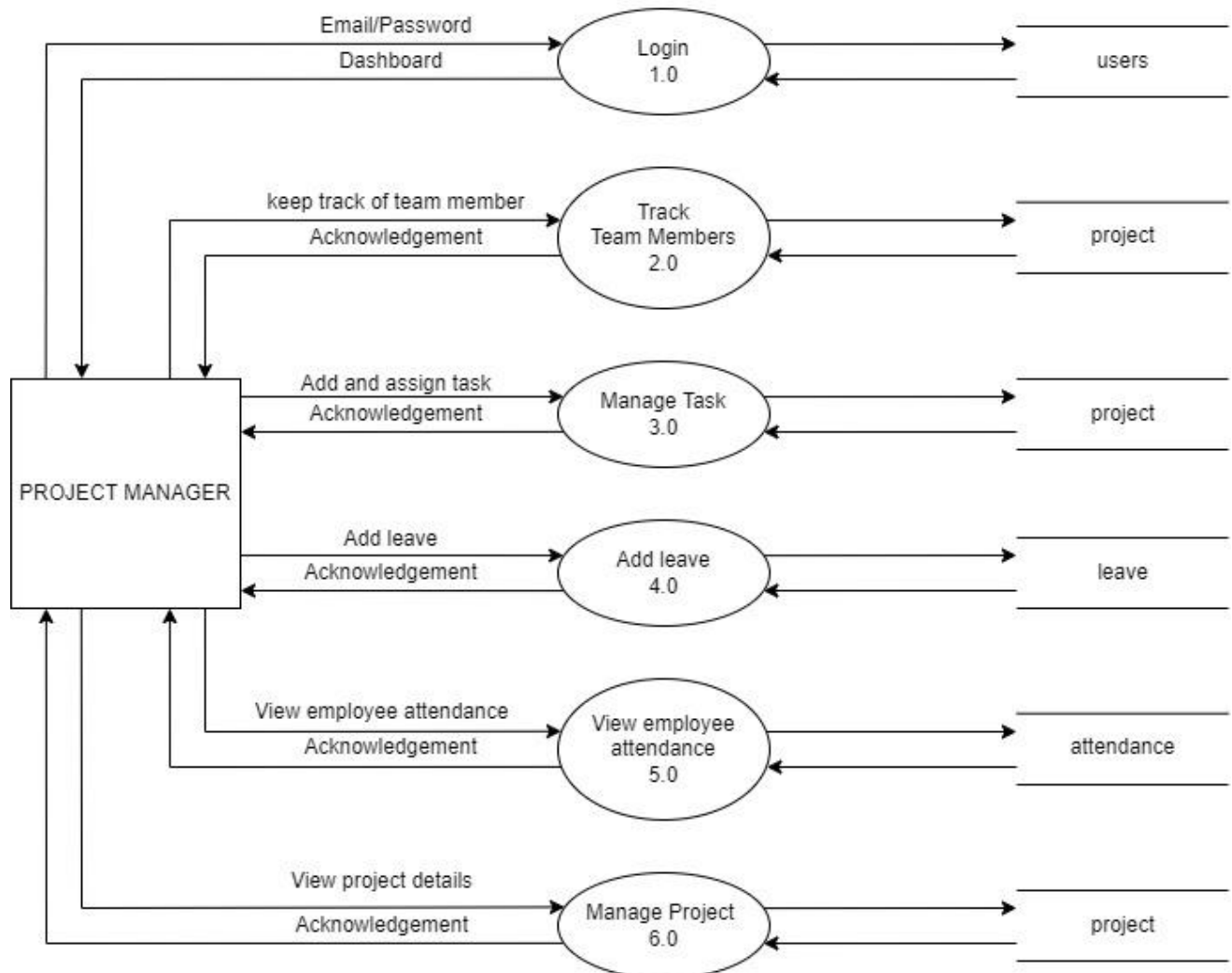
[Figure 3: DFD Level 1: Admin]

Level 1: HR



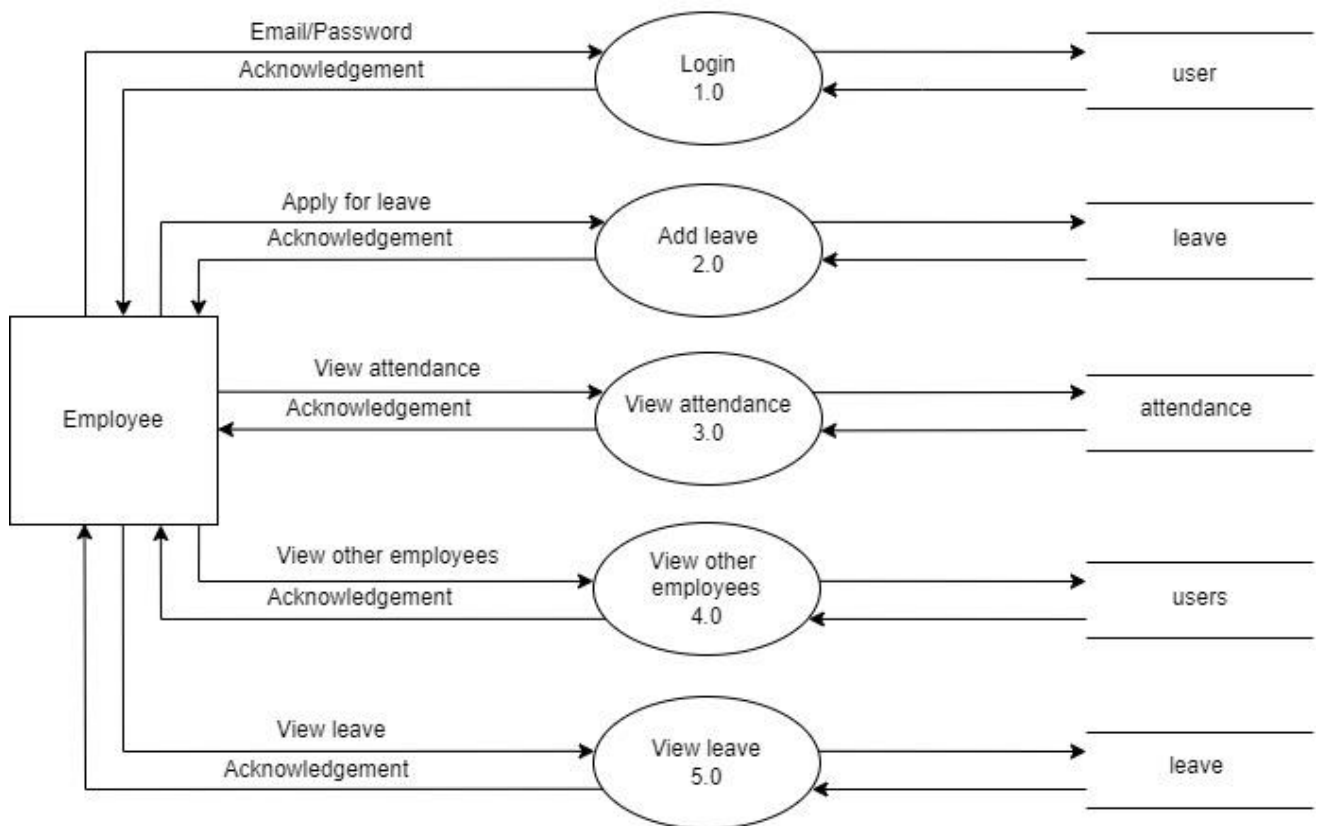
[Figure 4: DFD Level 1: Human Resource]

Level 1: PROJECT MANAGER



[Figure 5: DFD Level 1: Project Manager]

Level 1: EMPLOYEE


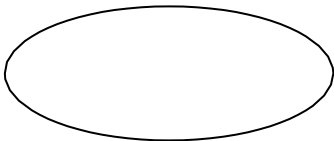
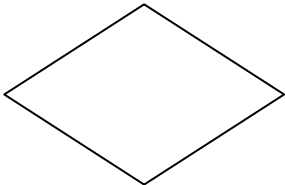
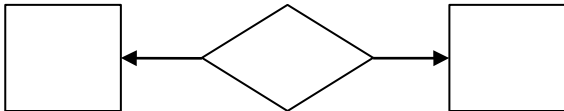
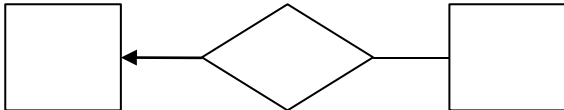
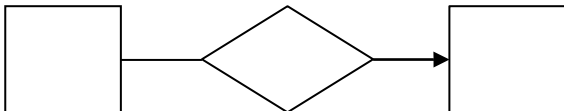
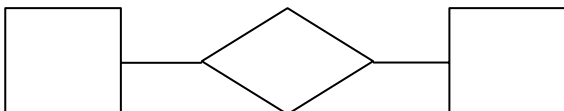


[Figure 6: DFD Level 1: Employee]

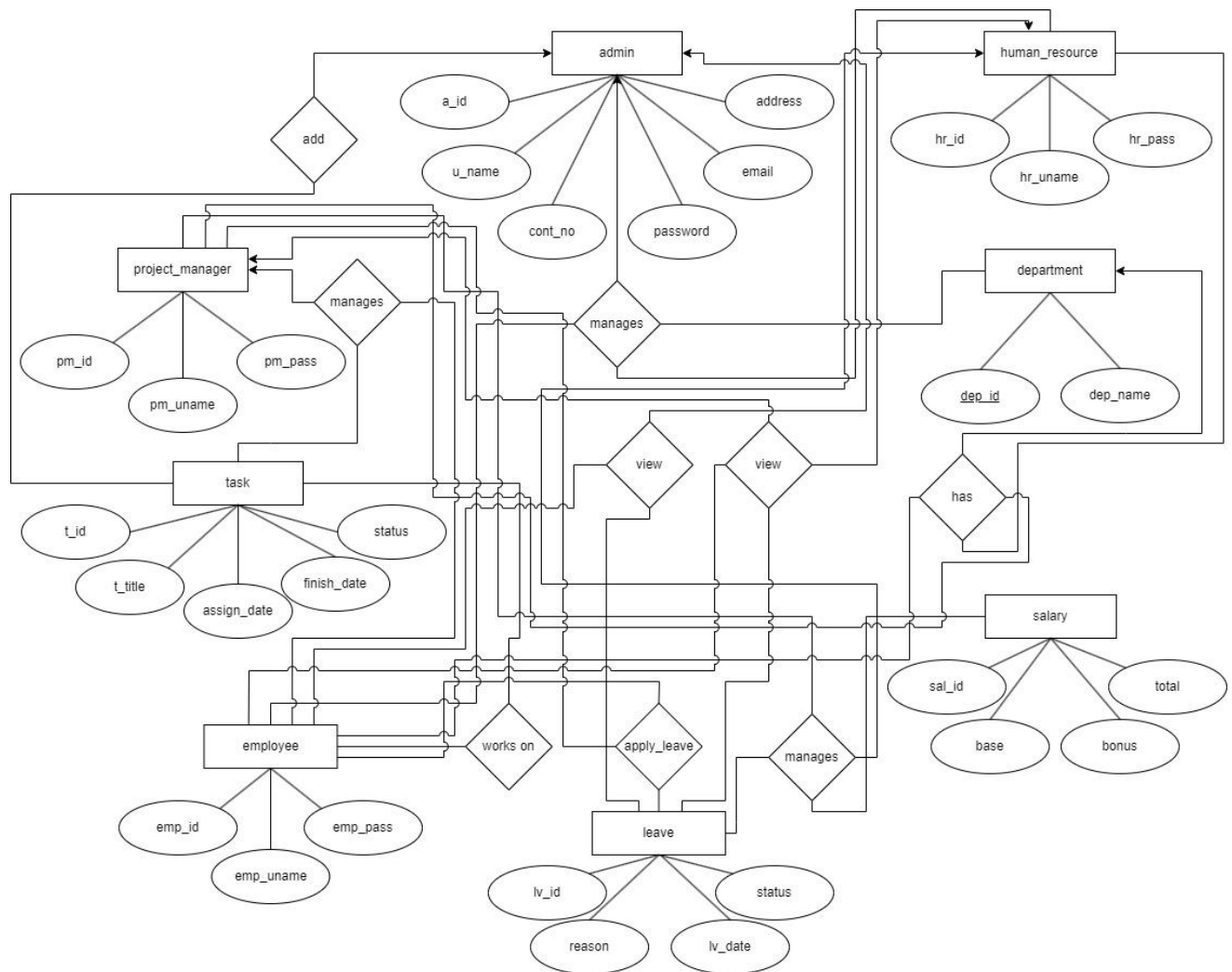
3.2 ER-Diagram

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

[Table 3: ER-Diagram Symbols]

Symbols	Description
	Entity: Data object is real world entity or thing. It is represented by a rectangle shape. An entity is an object or concept about which you want to store information.
	Attributes: An attribute is property of characteristic of an entity. It is represented by oval shape.
	Relationship: Entity are connected each other via relations. Generally, relationships in binary because there are two entities are related to each other.
	Cardinality (One to One): An instance of entity A can relate to one instances of entity B.
	Cardinality (One to Many): An instance of entity A can relate to one or many instances of B but we can only relate one instance of A.
	Cardinality (Many to One): One or more instances of entity A can relate to one instances of B.
	Cardinality (Many to Many): One or more instances of entity A can relate to one more instance of entity B.

ER-DIAGRAM:



[Figure 7: ER Diagram]