Figure: Use Case Diagram

Activity Diagram For User login



Activity Diagram For Admin Login



Activity Diagram For Show The Product List



Activity Diagram for Managing Bill



Activity Diagram for managing Category Of The product



Activity Diagram For Managing Feedback



Activity Diagram For Generate Sales Report



Swim lane Diagram

|  |  |  |
| --- | --- | --- |
| Customer | System | Admin |
|  | Yes  No  In stock |  |

**6.3 Entity Relationship Diagram**

The Entity Relationship Diagram (ERD) enables a software engineer to specify the data objects

That are input and output from a system, the attributes that define the properties of these objects

And their relationship. It provides an excellent graphical representation of the data structures and

Relationship.

While useful for organizing data that can be represented by a relational structure, an entity relationship

Diagram can't sufficiently represent semi-structured or unstructured data, and an ERD

Is unlikely to be helpful on its own in integrating data into a pre-existing information system.

Three main components of an ERD are the entities, which are objects or concepts that can have

Data stored about them, the relationship between those entities, and the cardinality which defines

That relationship in terms of number.

**Relationship Cardinality:** Relationship cardinality refers to the number of entity instances

Involved in the relationship. The cardinality ratios are:

1: 1 (One to One)

1: N (One too Many)

M: N (Many to Many)

**Attributes:** Attributes are properties of entities. Attributes are represented by means of eclipses.

Every eclipse represents one attribute and is directly connected to its entity (rectangle).

**Primary Key**: A primary key is an attribute or collection of attributes that allow us to identify an

entity uniquely.

**Foreign key**: A foreign key is an attribute of a relation, which refers to an existing attribute of

another relationship.

****

**Figure: Entity Relationship Diagram**

**6.4 Data Flow Diagram:**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an

information system, modeling its process aspects. A DFD is often used as a preliminary step to

create an overview of the system, which can later be elaborated.

**Context Level Diagram:**

A System Context Diagram (SCD) in software engineering and systems engineering is a diagram

that defines the boundary between the system, or part of a system, and its environment, showing

the entities that interact with it. Context level diagram of my system is given below-

**Figure 6.10** Context Level Diagram

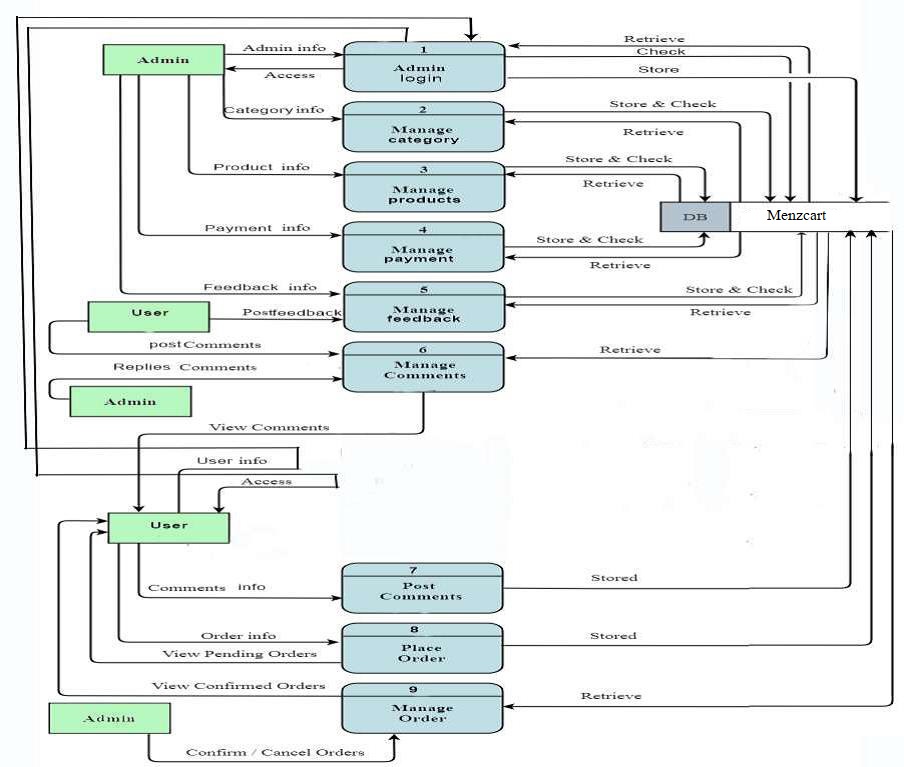
****

Figure: Level 1 DFD

Level 2 Process 1 DFD Login:



**Figure 6.12** Level 2 DFD of Process 1

Level 2 process 2 DFD (manage category)



Level 2 process 3 DFD (manage Product)



Level 2 process 4 DFD (manage Payment)



Level 2 process 5 DFD (manage Order)



Level 2 process 6 DFD (manage Comment)



Level 2 process 7 DFD (View Post)



Level 2 process 8 DFD (View Product)

