# ASSIGNMENT-1

## ART GALLERY MANAGEMENT

### DATABASE MANAGEMENT SYSTEM


**Submitted by**

*DEEKSHITHA R*
*PES2UG19CS104*

*DEEPA SHREE C V*
*PES2UG19CS105*

***Department of Computer Science and Engineering***
***Pes University***

## Project Title:

Art Gallery Management system.

## Problem Statement:

To create an efficient database management system for an Art Gallery.

## Introduction:

This project primarily deals with managing details of paintings, employees, customers and other stakeholders. We can maintain a record of the paintings present in the gallery, detailsof customers who buy these paintings. We can also enter new paintings available for sale. We can also record the details of the employees who work there.

## Objectives:

- To eliminate the paper work required in maintaining the records of sales and conducting exhibitions
- To help keep track of the employee details of the gallery
- To help recognize potential customers to increase the income of the gallery and also to help them connect with artists

## Assumptions:

- The Mini World considered is 1 single gallery.
- A painting is put up for auction only once in an exhibition.
- An employee must be at least 18 years old.

**Entities and Attributes (with the relevant constraints):**

- **Artist**

Attributes:

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| A_Id(Primary Key) | Varchar(30) | Yes | Yes | 'None' |
| A_Name(Fname,Mname,Lname) | Varchar(30) | Yes(MName can be Null) | Yes | 'None' |
| A_Email | Varchar(30) | Yes | Yes | 'None' |
| A_Ph_no | int | yes | Yes | 0000000000 |
| A_Gender | Varchar(6) | No | No | 'Not Mentioned' |
| A_Address | Varchar(100) | NO | No | 'None' |

- **Customer**

Attributes:

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| C_Id(Primary Key) | Varchar(30) | Yes | Yes | 'None' |
| C_Name(Fname,Mname,Lname) | Varchar(30) | Yes(MName can be Null) | Yes | 'None' |
| C_Email | Varchar(30) | Yes | Yes | 'None' |
| C_Ph_no | int | yes | Yes | 0000000000 |
| C_Gender | Varchar(6) | No | No | 'Not Mentioned' |
| C_Address | Varchar(100) | No | No | 'None' |

- **Paintings**

Attributes:

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| P_Id(Primary Key) | Varchar(30) | Yes | Yes | 'None' |
| P_Name | Varchar(30) | Yes | Yes | 'None' |
| P_Price | float | yes | no | 0.0 |
| A_Id(Foreign Key referencing to Artitst A_id) | Varchar(30) | yes | No | 'None' |
| C_Id(Foreign Key referencing to Customer C_id) | Varchar(30) | yes | No | 'None' |

- **Employee**

Attributes:

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| E_Id(Primary Key) | Varchar(30) | Yes | Yes | 'None' |
| E_Name(Fname,Mname,Lname) | Varchar(30) | Yes(MName can be Null) | Yes | 'None' |
| Date_of_join | Date | Yes | No | 'NA' |
| E_Ph_no | int | yes | Yes | 0000000000 |
| E_Gender | Varchar(6) | No | No | 'Not Mentioned' |
| E_DOB | Date | yes | No | Current Date |
| E_Age(Derived) | int | yes | No | 0 |
| D_No(Foreign key referencing to Department D No) | int | yes | No | 0 |
| E_Salary | float | Yes | No | 0.0 |
| E_Address | Varchar(100) | No | No | 'None' |

- **Department**

Attributes:

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| D_no(Primary Key) | Int (2) | Yes | Yes | 0 |
| D_Name | Varchar(40) | Yes | Yes | 'None' |
| Head_Id | Varchar(20) | yes | no | 0.0 |
| A_Id(Foreign Key referencing Employee E_id) | Varchar(30) | Varchar(30) | Yes | 'None' |
| C_Id(Foreign Key referencing to Customer C_id) | Varchar(30) | yes | No | 'None' |

- **Exhibition**

Attributes:

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| Ex_Id(Primary Key) | Varchar (20) | Yes | Yes | 'None' |
| Ex_Start_date | Date | Yes | Yes | Current Date |
| Ex_End_date | Date | Yes | Yes | Current Date |
| Ex_Name | Varchar (30) | Varchar(30) | Yes | 'None' |

- **Auction (Weak Entity)**

Attributes: (The combination of these keys is always unique)

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| Ex_Id(Foreign Key Referencing Exhibition Ex_Id) | Varchar (20) | Yes | No | 'None' |
| P_Id(Foreign Key Referencing Painting P_Id) | Varchar (30) | Yes | No | 'None' |

- **Instalments (Weak Entity)**

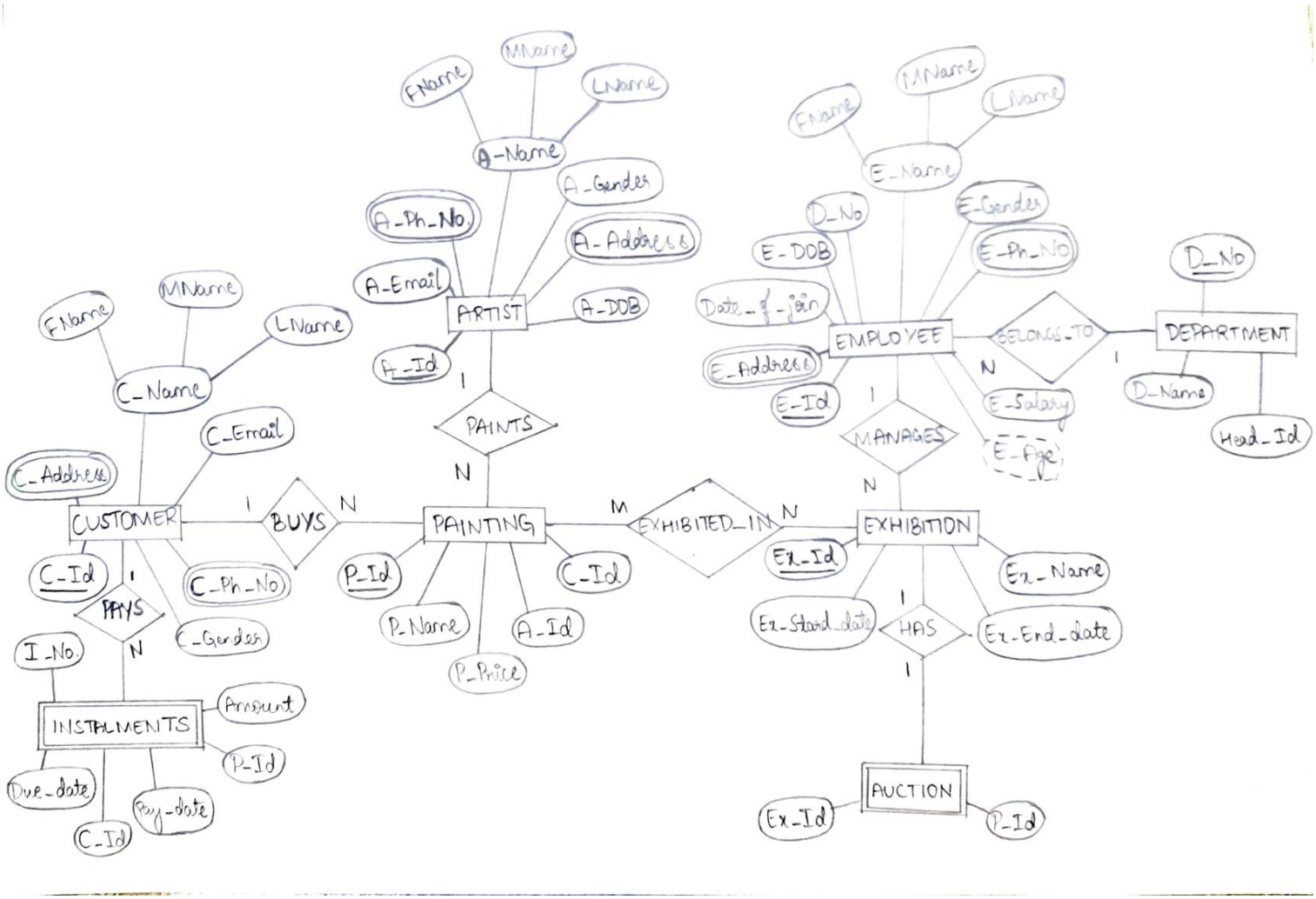
Attributes: (The combination of P_Id, C_Id and I_no is the unique key of this relation)

| Key | Datatype | NOT NULL | UNIQUE | Default |
|---|---|---|---|---|
| I_no(Instalment Number) | int | Yes | No | 0 |
| P_Id(Foreign Key referencing Painting P_Id) | Varchar(30) | Yes | No | 'None' |
| Pay_date | Date | Yes | No | Current Date |
| Due_date | Date | Yes | No | Current Date |
| Amount | float | yes | No | 0.0 |
| C_Id(Foreign Key referencing to Customer C_id) | Varchar(30) | yes | No | 'None' |

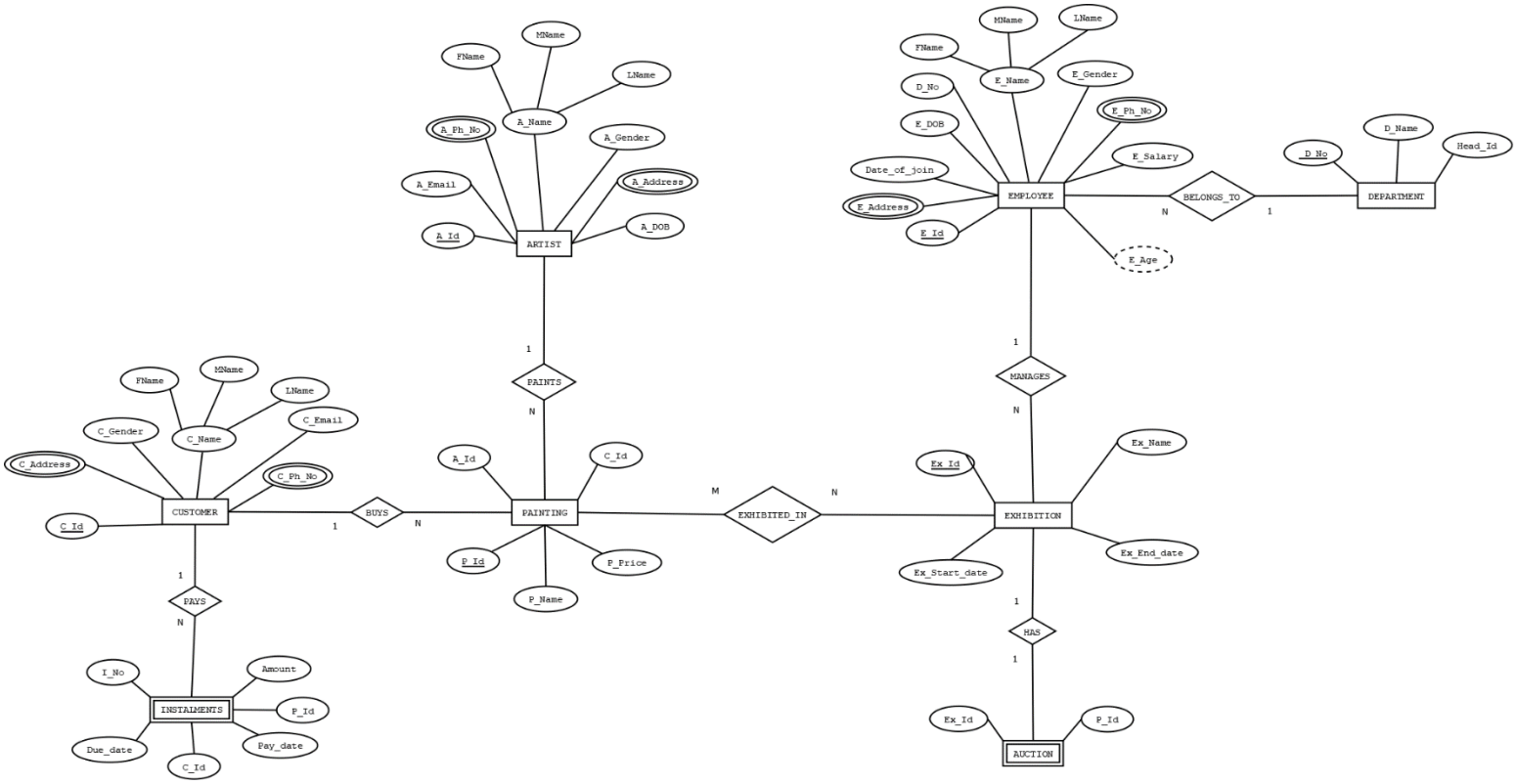
**Relationships:**

- **PAINTS**(Artist PAINTS Painting)
  It has a Cardinality Ratio of 1:N because 1 artist can paint many (N) paintings but1painting can be painted by one(1) artist only.
- **BUYS** (Customer BUYS Painting)
  It has a Cardinality Ratio of 1:N because 1 customer can buy many (N) paintings but1 painting can be sold to only 1 customer.
- **EXHIBITED_IN** (Painting EXHIBITED_IN Exhibition)
  It has a Cardinality Ratio of M:N because 1 painting can be exhibited in many(N)exhibitions and one exhibition can hold many(M) paintings.
- **MANAGES** (Employee MANAGES Exhibition)
  It has a Cardinality Ratio of 1:N because 1 Manager can manage many (N)exhibitions but 1 exhibition can have 1 Manager only.
- **PAYS** (Customer PAYS Instalments)
  It has a Cardinality Ratio of 1:N because 1 Customer can pay many(N)Instalments but the 1 instalment is paid by only 1 customer.
- **BELONGS_TO**(Employee BELONGS_TO Department)
  It has a Cardinality Ratio of N:1 because N employees can belong to 1 department but 1 employee can't be a part of multiple departments.
- **HAS**(Exhibition HAS an Auction)
  It has a Cardinality Ratio of 1:1 because 1 Exhibition has only 1 Auction and 1 Auction can be hosted at 1 Exhibition at a time.

## ER-Diagram (HAND-DRAWN):


FName
MName
LName
A_Name
A_Gender
A_Ph_No.
A_Address
A_Email
ARTIST
A_DOB
A_Id
1
PAINTS
N
FName
MName
LName
C_Name
C_Email
C_Address
CUSTOMER
1
BUYS
N
C_Id
1
PAYS
C_Ph_No
C_Gender
N
I_No.
INSTALMENTS
Amount
P_Id
Due_date
Pay_date
C_Id
P_Id
P_Name
A_Id
P_Price
C_Id
PAINTING
M
EXHIBITED_IN
N
FName
MName
LName
E_Name
E_Gender
D_No
E_DOB
E_Ph_NO
Date_of_join
E_Address
E_Id
EMPLOYEE
1
MANAGES
N
N
BELONGS_TO
1
E_Salary
E_Age
D_NO
DEPARTMENT
D_Name
Head_Id
Ex_Id
Ex_Name
Ex_Stard_date
1
HAS
Ex_End_date
1
EXHIBITION
Ex_Id
AUCTION
P_Id

## ER Diagram:




## Tool Used for the ER diagram:

Dia was used for the modelling of the ER Diagram. It is a general-purpose diagramming software, developed by Alexander Larsson. It has modular design with several packages available for different needs like flowchart, network diagrams, circuit diagrams and more. It has special objects to help draw ER models, UML diagrams, flowcharts, network diagrams. It also loads and saves diagrams in a custom XML format and can easily be exported to a PNG file. It is a **free** and **open-source** software. Link to download the software: here.

Steps for installation:

- Download the executable file from the above-mentioned website.
- Open the executable file, a popup window is shown.
- Select the desired language.
- Click “Next” until the final page is reached.
- Click “Install” to install the required modules.
- Click on “Finish” to install the software.

## Contributions:

**Deekshitha**: ER Diagram (both soft copy and hand drawn); Finalising the problem statement and entities, attributes
**Deepa**: Finalising the problem statement and entities, attributes; Report write-up, Relations for the database; Constraints for the Attributes