Chittagong University of Engineering and Technology

Department of Computer Science and Engineering

Project Title: Art Gallery Management System

Course Title: Database Management System (Sessional)

Course No.: CSE-352

Submission Date: 08/11/2017

Submitted To: Rahma Bintey Mufiz Mukta
Assistant Professor , Dept. of CSE, CUET.

Submitted By: Sumnoon Ibn Ahmad
ID: 1404006

Yasir Mahabub
ID: 1404014

Mahidul Islam Khandakar
ID: 1404024

Abstract

This project has been built on the desire of fulfilling the recreational need of a person. After fulfilling the basic needs, people want to do something to fulfil his mind. For example, when he has a house he wants to decorate it with flower, art etc. This project will keep a database on various types of artworks that will help a person to choose from different types of categories such as Abstracts, History Paintings, Landscape Paintings, Still Life Paintings, Sculpture, Calligraphy, Photography etc.

The use of a Database Management System (DBMS for short) for painting or artwork archiving is a reasonable choice. With the aid of an ordinary DBMS it is possible to store, modify, display painting records as well as find records that satisfy certain search criteria.

Although commercial database products do offer this level of functionality, they are not specialized to painting/artwork archiving. Therefore, special software must be developed on top of a DBMS that satisfies the needs of artwork archiving, study and conservation.

In this project, a database will contain information about “Artist” i.e. their name, birthplace, age, style of art. It will also contain information of Art like the class in which this artwork can be classified, its background history etc. Also like a commercial database, it will let people to buy the artwork for a certain price. It will also keep information about Customers as their ID, address, total amount of dollar they spent on Gallery and liking of Customers.

We will use primarily SQL for database management. For front end of the website, HTML and CSS will be used.PHP will be used for connecting Front end and database.

Table of Content

Name	Page
1. Abstract	1
2. Content	2
3. Introduction	3
4. ER Diagram	6
5. Relational Schema	12
6. Normalization	18
7. Implementation	24
8. Conclusion	37

Chapter 1

Introduction

A database is an organized collection of data.A relational database, on the other hand, is a collection of schemas, tables, queries, reports, views, and other elements. Database designers typically organize the data to model aspects of reality in a way that supports process requiring information, such as (for example) modelling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

A database-management system (DBMS) is a computer-software application that interacts with end-users, other applications, and the database itself to capture and analyze data. A general-purpose DBMS allows the definition, creation, querying, update, and administration of databases. Well-known DBMSs include MySQL, PostgreSQL, EnterpriseDB, MongoDB, MariaDB, Microsoft SQL Server, Oracle, Sybase, SAP HANA, MemSQL, SQLite and IBM DB2.Some application of database management system are ,

- **Telecom :** There is a database to keeps track of the information regarding calls made, network usage, customer details etc. Without the database systems it is hard to maintain that huge amount of data that keeps updating every millisecond.

- **Industry :** Where it is a manufacturing unit, warehouse or distribution centre, each one needs a database to keep the records of ins and outs. For example distribution centre should keep a track of the product units that supplied into the centre as well as the products that got delivered out from the distribution centre on each day; this is where DBMS comes into picture.

- **Banking System :** For storing customer info, tracking day to day credit and debit transactions, generating bank statements etc. All this work has been done with the help of Database management systems.

- **Education Sector :** Database systems are frequently used in schools and colleges to store and retrieve the data regarding student details, staff details, course details, exam details, payroll data, attendance details, fees details etc. There is a hell lot amount of inter-related data that needs to be stored and retrieved in an efficient manner.

- **Online shopping :** We are aware of the online shopping websites such as Amazon, Flipkart etc. These sites store the product information, your addresses and preferences, credit details and provide us the relevant list of products based on our query. All this involves a Database management system.

Some Key Terms :

Here we will discuss about some key terms that is used for discussing different steps of creating a database management system,

1. Entity Relationship Diagram:

   An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structure of databases. At first glance an entity relationship diagram looks very much like a flow-chart. It is the specialized symbols, and the meanings of those symbols, that make it unique.

2. Relational Schema :

   The term “schema” refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database.

3. Database Normalization :

   Database normalization, or simply normalization, is the process of organizing the columns (attributes) and tables (relations) of a relational data-

base to reduce data redundancy and improve data integrity. Normalization is also the process of simplifying the design of a database so that it achieves the optimal structure composed of atomic elements. It was first proposed by Edgar F. Codd, as an integral part of a relational model.

4. SQL :

   SQL is a programming language for Relational Databases. It is designed over relational algebra and tuple relational calculus. SQL comes as a package with all major distributions of RDBMS.

There are more terms in database. Those will be discussed in different sections .

Specification :

Category: In this website, the artworks will be categorized by year, artist, genre, price range and area. So people can choose the arts by their choice. Also asearch option will be added so that visitors can easily find their desired item.

Background Details: Details of the artist and artwork will be given so that people can know more about the artist and his/her artwork. Also, they can learn the background of the artwork: why it was drawn, what it symbolizes, it's influence on history so that people can understand its significance.

Availability: Anyone can visit the website but if they want to buy or give reviews, they will have create an account.

Buying Option: Customers will be able to purchase artworks that are available for sale by bKash. They will be able to buy multiple artworks at a time and can keep track of which artworks they have bought and how much money they spent.

Review System: Customers will also be able to give reviews on particular artworks. There will be also a star review system.

Favorite Option: There will be an option for the customer to like a particular artwork so that he/she can easily find it the next time he/she visits.

Chapter 2

ER DIAGRAM

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structure of databases.

At first glance an entity relationship diagram looks very much like a flowchart. It is the specialized symbols, and the meanings of those symbols, that make it unique.

When documenting a system or process, looking at the system in multiple ways increases the understanding of that system. ERD diagrams are commonly used in conjunction with a data flow diagram to display the contents of a data store.

They help us to visualize how data is connected in a general way, and arepartic-ularly useful for constructing a relational database.

An ER diagram is a means of visualizing how the information a system produces is related. There are five main components of an ERD:

- **Entities**, which are represented by rectangles. An entity is an object or concept about which you want to store information.

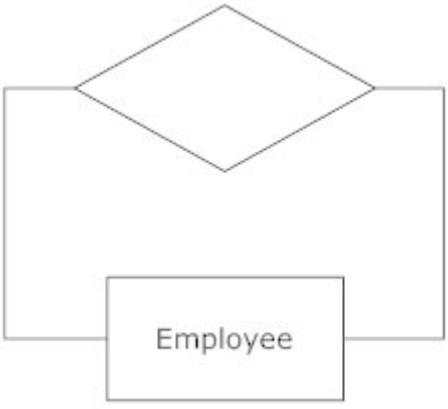
  A weak entity is an entity that must defined by a foreign key relationship with another entity as it cannot be uniquely identified by its own attributes alone.

In our Art Gallery Management System there are three entities ,

1. Customer
2. Artwork
3. Artist

- **Actions**, which are represented by diamond shapes, show how two entities share information in the database.

In some cases, entities can be self-linked. For example, employees can supervise other employees.



- **Attributes**, which are represented by ovals. A key attribute is the unique, distinguishing characteristic of the entity. For example, an employee's social security number might be the employee's key attribute.

A multivalued attribute can have more than one value. For example, an employee can have multiple skill values.

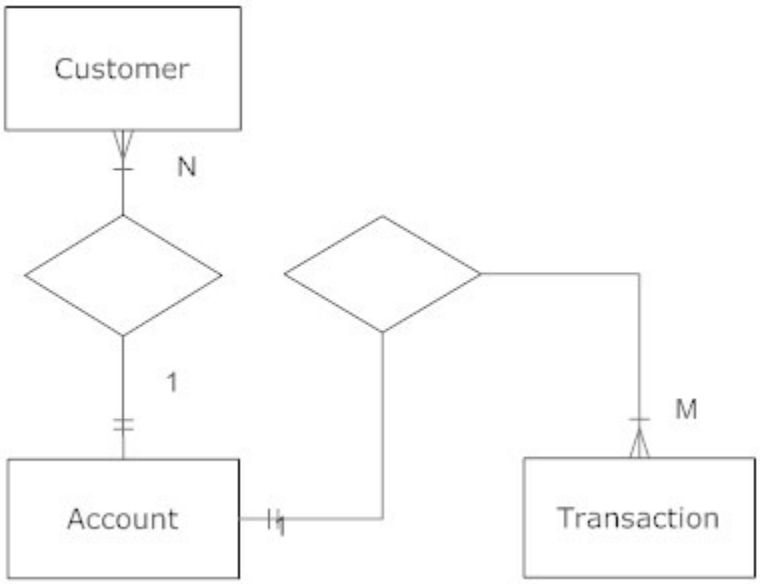
A derived attribute is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary.

In the next table we have shown the attributes of different entities of our project in tablular form

Entity No.	Entity Name	Attributes
1	Customer	Customer_Id, Customer_Name, User_ Name, Pwd, Address, Credit.
2	Art	Art_Id, Art_Name, Reason, Art_Influence, Price, Year, Location, Area, Aid, Cid.
3	Artist	Artist_Id, Artist_Name, BirthPlace, Age, Style, Influence, Work, Cid.

- **Connecting lines**, solid lines that connect attributes to show the relationships of entities in the diagram.

- **Cardinality** specifies how many instances of an entity relate to one instance of another entity. Ordinality is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinality describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinality specifies the absolute minimum number of relationships.



There are some basic steps in creating an ER diagram these are

1. Identifying the entities.
2. Describing the relationship.
3. Adding the attributes.
4. Completing the diagram.

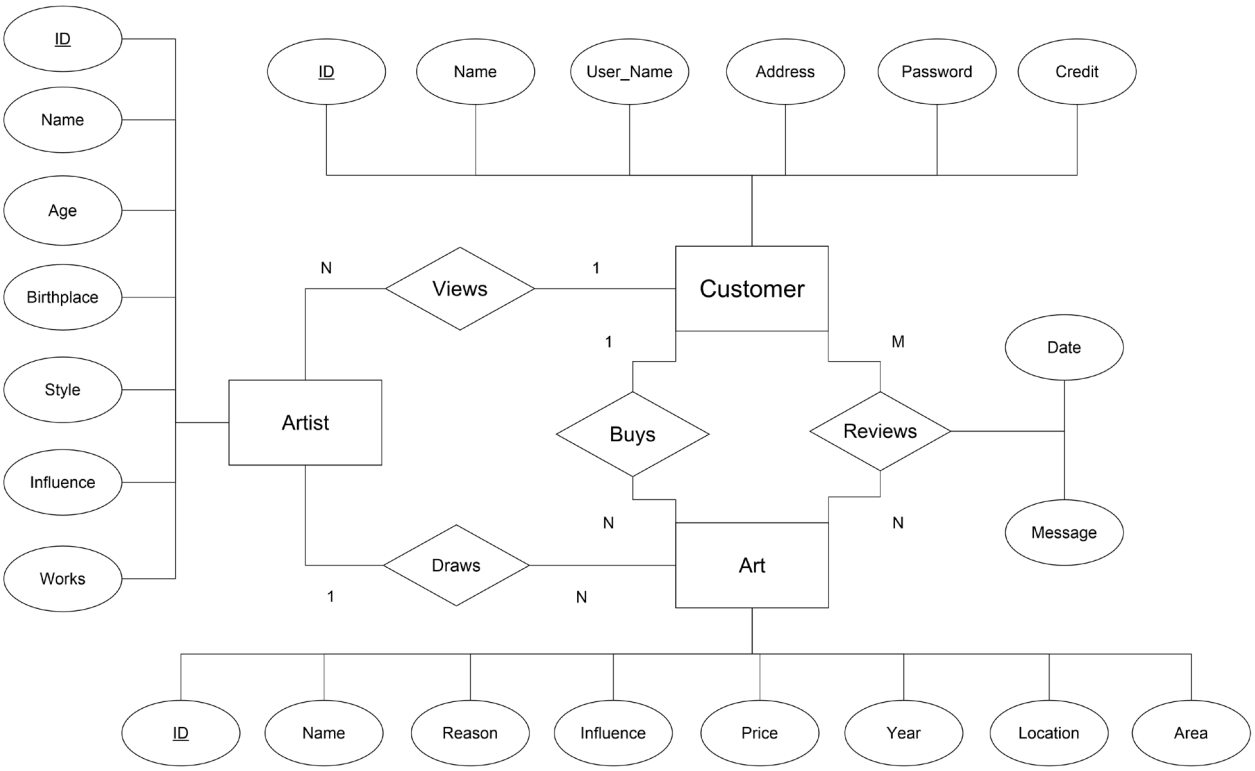


Figure: ER Diagram of ArtGallery Management System

Here the main entities are

Artist

Art

Customer

There are four relations.

- **Artist and Customer (One relation) :**

  Customer can learn about artist.As they visit the site.A customer can visit many art, so he/she do want to know about the artist who drew them.So this is an 1 to N relation.

- **Customer and Art (Two relations) :**

  There are two relations between customer and art

  One is customer can enjoy the art.Also if he/she want he/she can buy pictures.

  He/She can buy 1 or more pictures at a time.So its an 1 to N relation between

  customer and art.

  Another one is any customer or user can review a picture. As many customers can review many pictures it is an N to M relation.

- **Artist and Art (One relaton) :**

  The relation between art and artist are 1 to 1 because there is only one

  artist for each art.

Chapter 3

Relatonal Mapping

ER Model, when conceptualized into diagrams, gives a good overview of entity-relationship, which is easier to understand. ER diagrams can be mapped to relational schema, that is, it is possible to create relational schema using ER diagram. We cannot import all the ER constraints into relational model, but an approximate schema can be generated.

The direction of a relationship may be either unidirectional or bidirectional. In a unidirectional relationship, only one entity bean has a relationship field that refers to the other. All TopLink relational mappings are unidirectional, from the class being described (the *source* class) to the class with which it is associated (the *target* class). The target class does not have a reference to the source class in a unidirectional relationship. In a bidirectional relationship, each entity bean has a relationship field that refers to the other bean.Through the relationship field, an entity bean's code can access its related object. To implement a bidirectional relationship (classes that reference each other), use two unidirectional mappings with the sources and targets reversed

We present the concepts of a general mapping algorithm. The algorithm has 7 steps:

Step 1: Mapping of regular (strong) entity types

Step 2: Mapping of weak (dependent) entity types

Step 3: Mapping of binary relationship type of cardinality ratio 1:1

Step 4: Mapping of binary relationship type of cardinality ratio 1:N

Step 5: Mapping of binary relationship type of cardinality ratio N:M

Step 6: Mapping of multivalued attributes

Step 7: Mapping of n-ary relationship types, n > 2

Description of different steps are given below

Step 1: Mapping of Regular Entity Types

1. For each regular (strong) entity type E in the ER schema, create a relation R that includes all the simple attributes (or simple components of composite attributes) of E.
2. Choose one of the key attributes of E as primary key for R.
3. If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.

Step 2: Mapping of Weak Entity Types

1. For each weak entity type W with owner entity type E, create a relation R that includes all simple attributes (or simple components of composite attributes) of W as attributes of R.
2. Include as foreign key attribute(s) in R the primary key attribute(s) of the relation(s) that corresponds to the owner entity type(s).
3. The primary key of R is the combination of the primary key(s) of the owner(s) and the partial key of the weak entity type W, if any.

Step 3: Mapping of Binary 1:1 Relationship Types

For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R. There are three possible approaches for mapping

1. Foreign Key approach: Choose one of the relations (say S) and include as foreign key in S the primary key of T (it is better to choose an entity type with total participation in R in the role of S).
2. Merged relation option: Merge the two entity types and the relationship into a single relation (possible when both participations are total).
3. Cross-reference or relationship relation option: Set up a third relation R for cross-referencing the primary keys of the two relations S and T representing the entity types.

Step 4: Mapping of Binary 1:N Relationship Types

1. For each regular binary 1:N relationship type R, identify the relation S that represent the participating entity type at the Nside of the relationship type.

2. Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R.
3. Include any simple attributes of the 1:N relation type as attributes of S.

Step 5: Mapping of Binary M:N Relationship Types

1. For each regular binary M:N relationship type R, create a new relation S to represent R.
2. Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S.
3. Also include any simple attributes of the M:N relationship type (or simple components of composite attributes) as attributes of S.

When Mapping Binary Relation,

1. Foreign key option is preferred for 1:1 and 1:N relationships, but cannot be used for M:N relationships.
2. Relationship relation option can be used for any cardinality ratio, but the primary key will be different:
    - Combination of both foreign keys for M:N
    - Either foreign key for 1:1
    - Foreign key in the N-side relation for 1:N
3. Attributes of relationship type are included in the relationship relation(for cross-referencing option), or in the relation that includes the foreign key (for foreign key option).

Step 6: Mapping of Multivalued attributes

1. For each multivalued attribute A, create a new relation R.
2. This relation R will include an attribute corresponding to A, plus the

primary key attribute K (as a foreign key in R) of the relation that represents the entity type that has A as an attribute.

3. The primary key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.

Step 7: Mapping of N-ary Relationship Types

1. For each n-ary relationship type R, where n>2, create a new relationship relation S to represent R.
2. Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types.
3. Also include any simple attributes of the n-ary relationship type (or simple components of composite attributes) as attributes of S.

According to these steps the ER schema of Art Gallery is given below,

Step 1.

There are three regular entity Artist, Art and Customer For three strong entity three tables are created. And for each entity a primary key was selected (Artist_Id,Art_Id,Customer_Id respectively). Since these are not composite keys so we did not need to simplify them .

Step 2.

Since there is no weak entity we do not need any mapping.

Step 3.

There is only one 1:1 relation, Artist to Art. Here we have use foreign key approach. We have take the primary key of artist and inserted it into Art table as foreign key.

Step 4.

There are two 1:N relations, ‘Artist to Customer’ and ‘Customer to Art’ as a customer can learn about one or more artists or arts so art and artist is on the N side.Thats why primary key of customer is inserted as foreign key into artist and art table.

Step 5.

There is an M:N relation between customer and art.A customer can review any art also an art can be reviewed by many customer that’s why it is an M:N relation.

Step 6.

There is no multivalued attribute.

Step 7.

There is no need to use step 7 of the algorithm as there is no N-ary relationship.

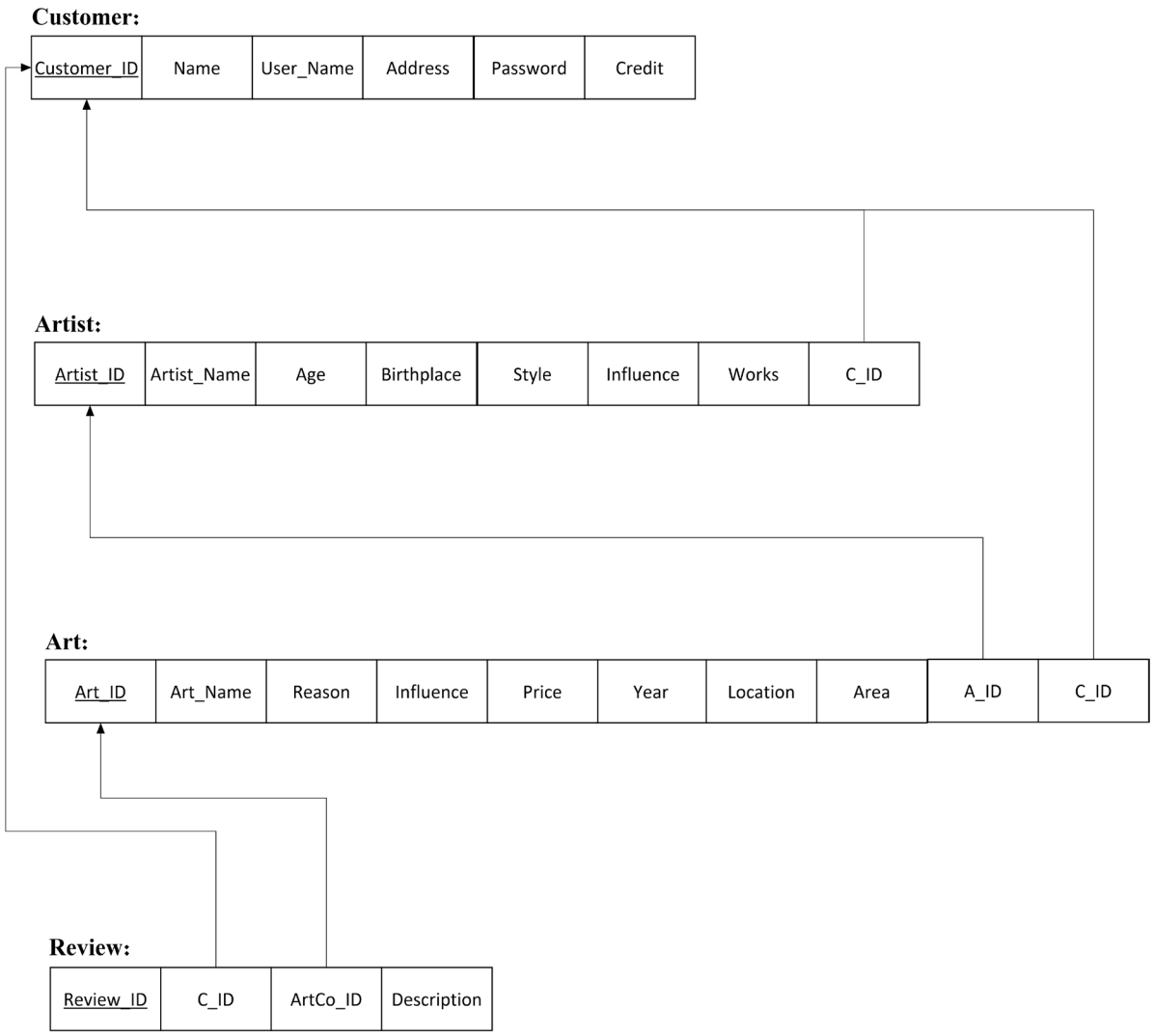


Figure: Relational Mapping of Art Gallery Management System

Chapter 4

Normalizaion

Normalization primalriy a tool to validate and improve a logical desing so that it satisfies certain constraints that avoid unnecessary duplication of data.It is a process of decomposing relations with anomalies to produce smaller well-structured relations.

The goal of normalization is to avoid anomalies , which are

1. Insertion Anomaly - Adding new rows forces user to create duplicate data.

2. Deletion Anomaly - Deleting rows may cause a loss of data that would be needed for other future rows.

3. Modification Anomaly - Changing data in a row forces changes to other rows because of duplication.

Normalization works through a series of stages called normal forms.These are Condition using keys and functional dependencies of a relation to certify whether a relation schema is in a particular normal form and their conditons are ,

i. First normal form (1NF)

1. No multivalued attributes

2. Every attribute value is atomic

ii. Second normal form (2NF)

The conditions are ,

1. It is in first normal form

2. It includes no partial dependencies; that is, no attribute is dependent on only a portion of the primary key.

iii. Third normal form (3NF)

The conditions are ,

1. It is in second normal form
2. It contains no transitive dependencies

Now before normalizing we have to know about functional dependencies,

We say an attribute, B, has a functional dependency on another attribute, A, if for any two records, which have the same value for A, then the values for B in these two records must be the same. We illustrate this as:

$$A \rightarrow B$$

For normalizing the art gallery DBMS lets at first look at the tables

Figure: Relational Mapping Before Normalization

From relational schemas we can normalize it easily

Customer Table :

Customer table has one Primary key that is customer_id.Normalization

procedure is given below

1. Converting to first normal form

The table is already in first normal form.Because before normalizing we have created the relations also we have created a single relation. For multivalued attribute if there is one.Also the composite attributes are already simplified so the table are in first normal form.

2. Converting to second normal form

The table is also in second normal form.Because the conditions for being second normal form is to be in first normal form and no partial dependencies.Since the relation is in first normal form and there is no composite primary key so there is no partial dependencies.

3. Converting to third normal form

There is no need to convert the relations into third normal form.As the relations are already in first and second normal form and there is no transitive dependencies.

Artist Table :

Artist table has one Primary key that is artist_id and one foreign key C_ID from customer table.Normalization procedure is given below.

1. Converting to first normal form:

The table is already in first normal form.Because before normalizing we have created the relation also there is no multivalued attribute. Also the composite attributes are already simplified so the table are in first normal form.

2. Converting to second normal form

The table is also in second normal form.Because the conditions for being second normal form is to be in first normal form and no partial dependencies.Since the relation is in first normal form and there is no composite primary key so there is no partial dependencies.

3. Converting to third normal form

There is no need to convert the relation into third normal form.As the relation are already in first and second normal form and there is no transitive dependencies.

Art Table :

Art table has one Primary key that is art_id and two foreign keys C_ID from customer table and A_ID from artist table.Normalization procedure is given below.

1. Converting to first normal form.

The table is already in first normal form.Because before normalizing we have created the relation also there is no multivalued attribute. Also the composite attributes are already simplified so the table are in first normal form.

2. Converting to second normal form

The table is also in second normal form.Because the conditions for being second normal form is to be in first normal form and no partial dependencies.Since the relation is in first normal form and there is no composite primary key so there is no partial dependencies.

3. Converting to third normal form

There is no need to convert the relation into third normal form.As the relation are already in first and second normal form and there is no transitive dependencies.

Comment Table :

Artist table has one Primary key that is comm_id and one foreign key ArtCo_Id from art table .Normalization procedure is

1. Converting to first normal form

The table is already in first normal form.Because before normalizing we have created the relation also there is no multivalued attribute. Also the composite attributes are already simplified so the table are in first normal form.

2. Converting to second normal form

The table is also in second normal form.Because the conditions for being second normal form is to be in first normal form and no partial dependencies.Since the relation is in first normal form and there is no composite primary key so there is no partial dependencies.

3. Converting to third normal form

There is no need to convert the relation into third normal form.As the relation are already in first and second normal form and there is no transitive dependencies.

After performing normalization procedure the normalized form of the ER Schema is given below

Customer:

Customer_ID	Name	User_Name	Address	Password	Credit

Artist:

Artist_ID	Artist_Name	Age	Birthplace	Style	Influence	Works	C_ID

Art:

Art_ID	Art_Name	Reason	Influence	Price	Year	Location	Area	A_ID	C_ID

Review:

Review_ID	C_ID	ArtCo_ID	Description

Figure: Normalized ER Schema

Chpater 5

Implimentation

SQL is a standard language for accessing and manipulating databases.

SQL stands for Structured Query Language. It is an ANSI (American National Standards Institute) standard. SQL can perform several manip- ulation on database. Some of them are following.

- Execute queries against a database
- Retrieve data from a database
- Insert records in a database
- Update records in a database
- Delete records from a database
- Create new databases
- Create new tables in a database
- Create stored procedures in a database
- Set permissions on tables, procedures, and views

Here we are discussing the implementation with brief explanetion

1. Creating the Front End :

For front end we have used html . We use CSS for designing the website.

So the total design of our website is like this,

Fig-1 Homepage1

2. Creating and Manipulating the database :

2.1 Table Creation

For manipulating data of different sections at first we have to create the database and tables for different entity and relations (if needed). The queries are,

Customer Table :

```
CREATE TABLE CUSTOMER (ID int(11) NOT NULL
        PRIMARY KEY AUTO_INCREMENT,NAME varchar(30)
        NOT NULL, USER_NAME varchar(30) NOT NULL,PASS-
        WORD varchar(1000) NOT NULL, CREDIT INT (12)
);
```

Artist Table :

```
CREATE TABLE `artist` (`Artist_Id` int(12) NOT NULL PRIMARY
KEY
        AUTO_INCREMENT, `Artist_Name` varchar(40) NOT
```

NULL, `BirthPlace` varchar(40) NOT NULL,`Influence` varchar(1000) NOT

NULL,`Work` varchar(500) NOT NULL,`Cid` int(12) NOT NULL

);

Art Table:

CREATE TABLE `art` (`Art_Id` int(12) NOT NULL PRIMARY KEY

AUTO_INCREMENT, `Art_Name` varchar(40) NOT NULL,

`Reason` varchar(1000) DEFAULT NULL,`Art_Influence` varchar(1000)DEFAULT NULL,`Price` int(15) NOT NULL,`Year` int(6) DEFAULT NULL,`Location` varchar(40) DEFAULT NULL,`Aid` int(12) NOT NULL DEFAULT '0','Cid` int(12) NOT NULL DEFAULT '0'

);

Comment Table:

CREATE TABLE comments (

Comm_Id int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,

uid varchar(128) not null,date datetime not null, message TEXT not

null

);

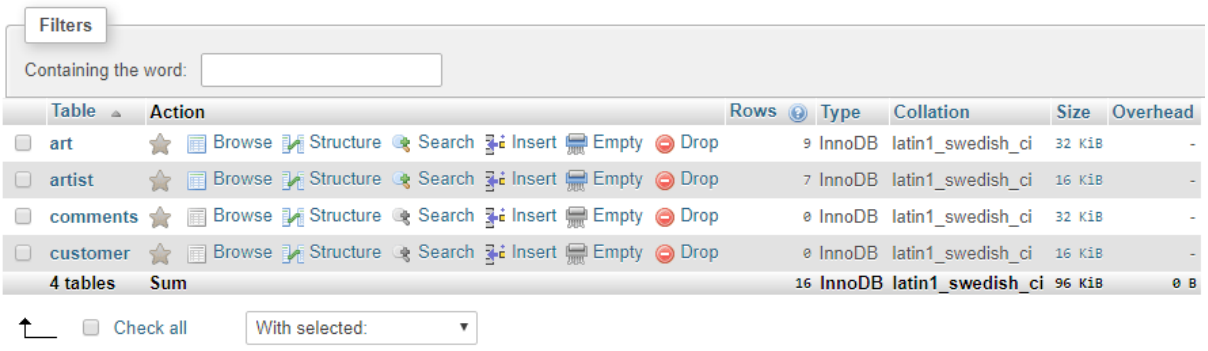


Fig-Created Table

2.2 Customer Section :

There are 4 types of queries in customer section.These are given below

Inserting customer data:

For signup process we need data from customer. For that reason a customer will give his name,user name and password for login process. Then data will be inserted into customer table.

For that sql query will be ,

INSERT INTO customer (Customer_name, User_name,P-wd,Credit)

VALUES (‘$name’, ‘$uname’, ‘$pwd’, 0);

Here $name,$uname,$pwd,0 are the values of Customer_name,

User_name, Pwd , Credit respectively.

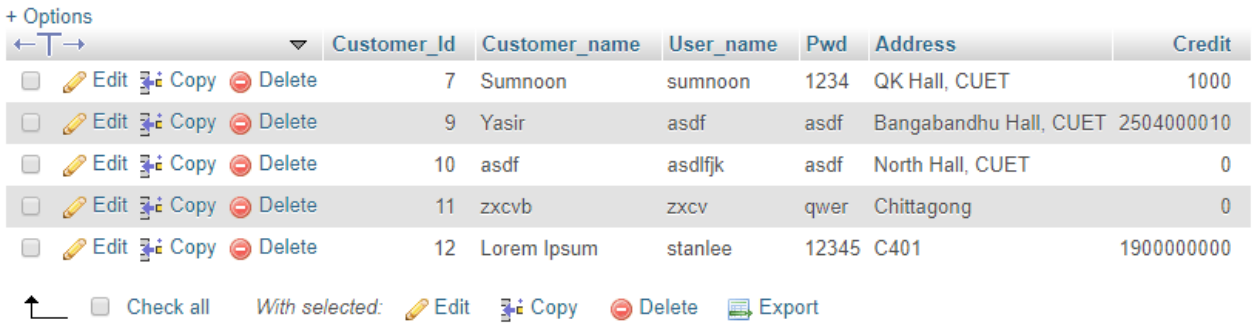


Fig-2 : Before SignUp

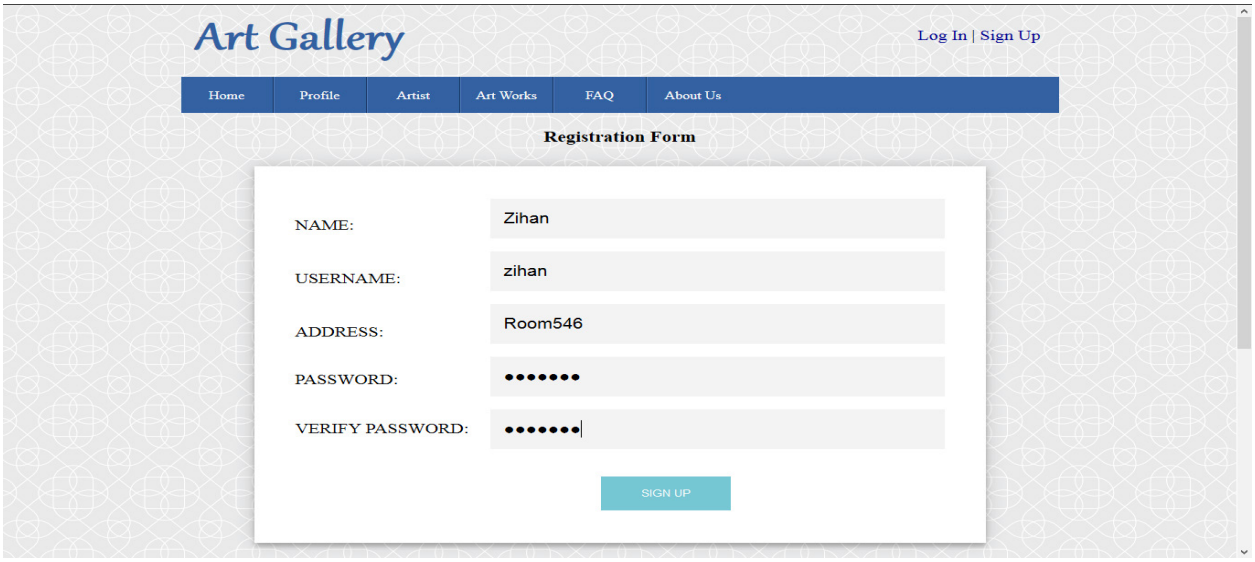


Fig-3 : SignUp Page

Fig-4 : After SignUp

Signing In:

For sign in a customer must have his data included in database. For

signing in when he gives his user name and password it will be searched in database than if matched he will be logged in other-wise error will show.

For searching sql query is,

SELECT * FROM customer WHERE User_name='$uname'

AND Pwd = '$pwd' ;

Here $uname and $pwd are supposed username and password.

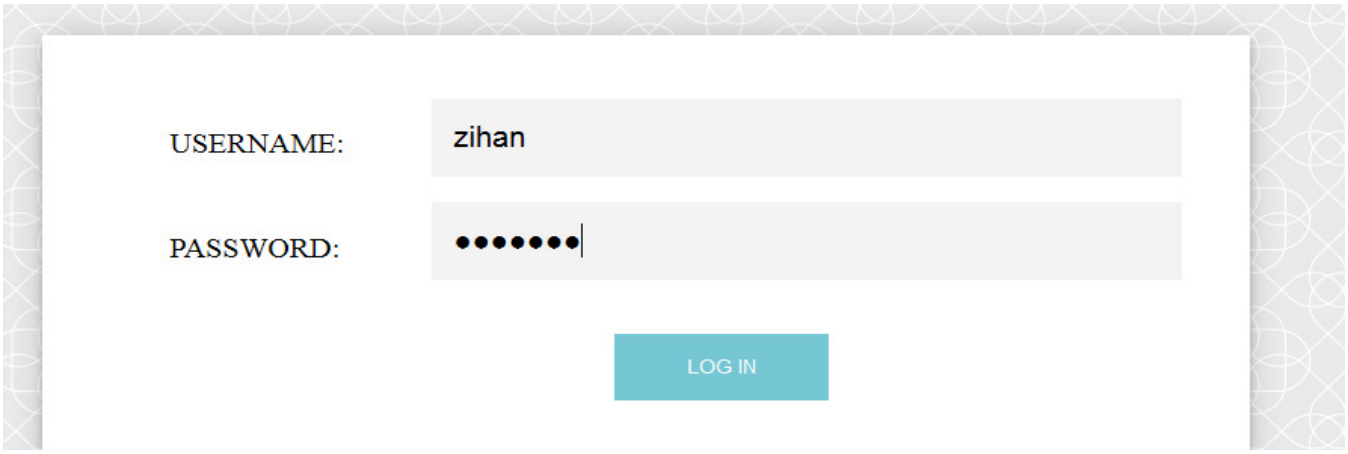


Fig-5 : Login Page

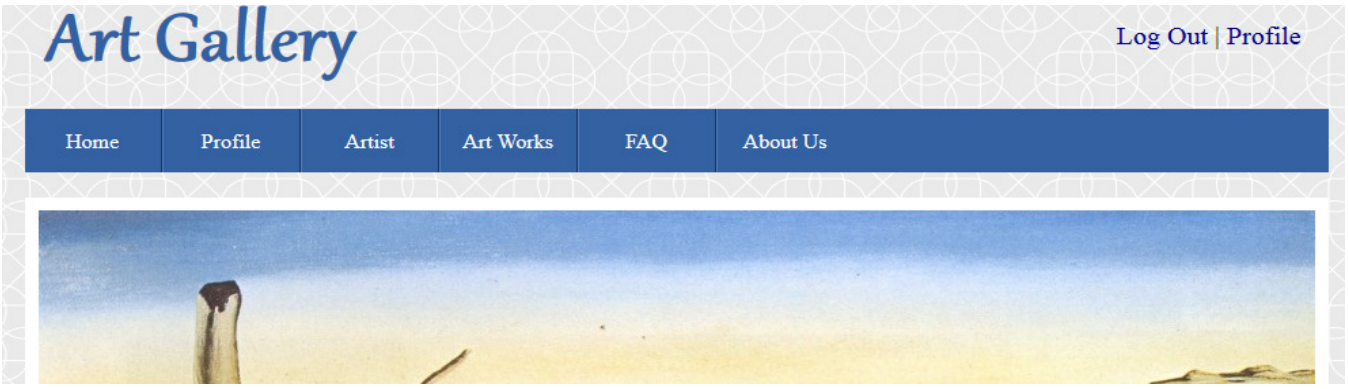


Fig-6 : After Log in

Deleting a user :

To delete a user profile one can click delete button then after confirmation the profile will be deleted.

SQL query for deletion is,

$sql = “DELETE FROM customer WHERE User_name=’$uname’”;

Here $uname is the user name of the customer.

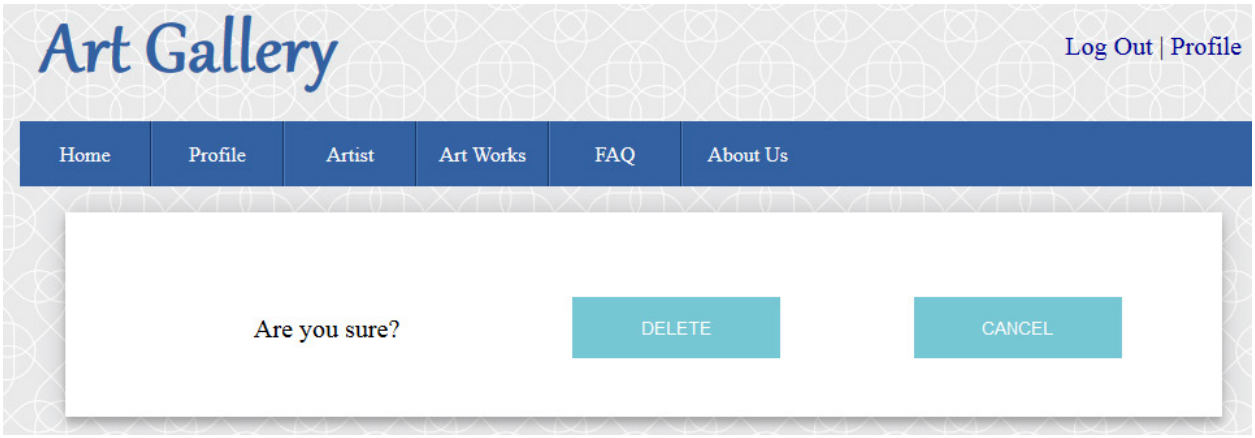


Fig-7 : Deleting User

Fig-8 : Customer Table after deleting

Editing a user profile :

To edit a user profile one can click edit button then he/she can edit his/her

Name and Password.

SQL query for editing is

$sql = “UPDATE customer SET Customer_name = ‘$name’

WHERE User_name=’$uname’”;

Here $uname is the user name of the customer.

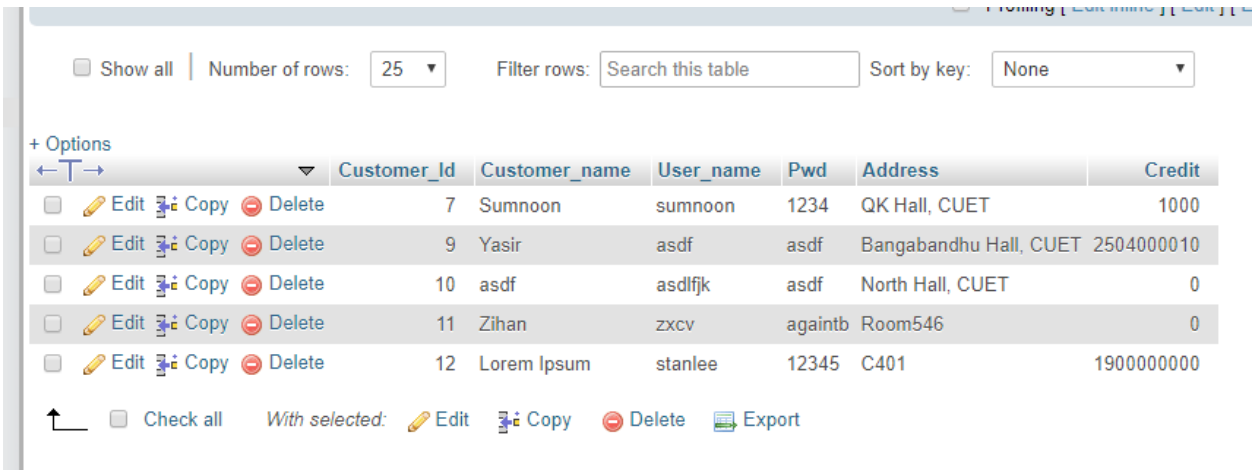


Fig-9 : Table after editing

2.3 Manipulation of Artist Section :

To know about any artist we need to click to Artist tab and go to artist list . There we can see all the artist. Then if we select name of any artist we will see details about the artist. The query for artist is

$sql = “SELECT * FROM artist WHERE Artist_Id=’$id’”;

Here $id is the desired artist id and when user inquire about the artist php will send the query to sql and it shows the whole data.

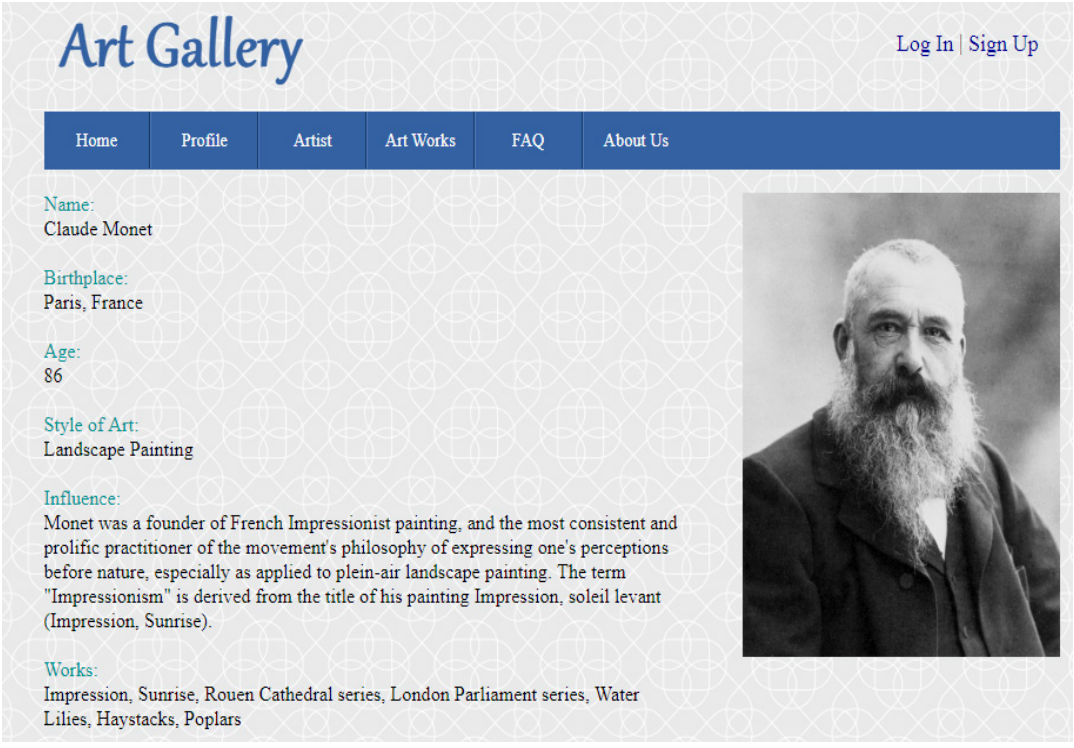


Fig-10 : Artist Page

2.4 Manipulating Art Work :

To manipulate art work we will use sql queries for different operation in art table.These operations are given below.

Showing data of an Art work :

If we want to see any art we need to click to Artwork tab and go to art list . There we can see all the art. Then if we select any artwork we will see details about this particular artwork. The query for artwork is

$sql = “SELECT * FROM art WHERE Art_Id=’$id’”;

Here $id is the desired art id and when user inquire about the art work, php will send the query to sql and it shows the whole data.

Showing Catagories of Art Work :

Also we can select different types of category like money range, region, year region etc. using sql queries.For example to find arts in between range from 1 to 1 Million we have to run the query,

$sql = “SELECT * FROM art WHERE Price BETWEEN 1 AND 1000000

ORDER BY Price, Art_Name”;

Using this query we can see the art list which has price between 1 to 1 Million in sorted list.

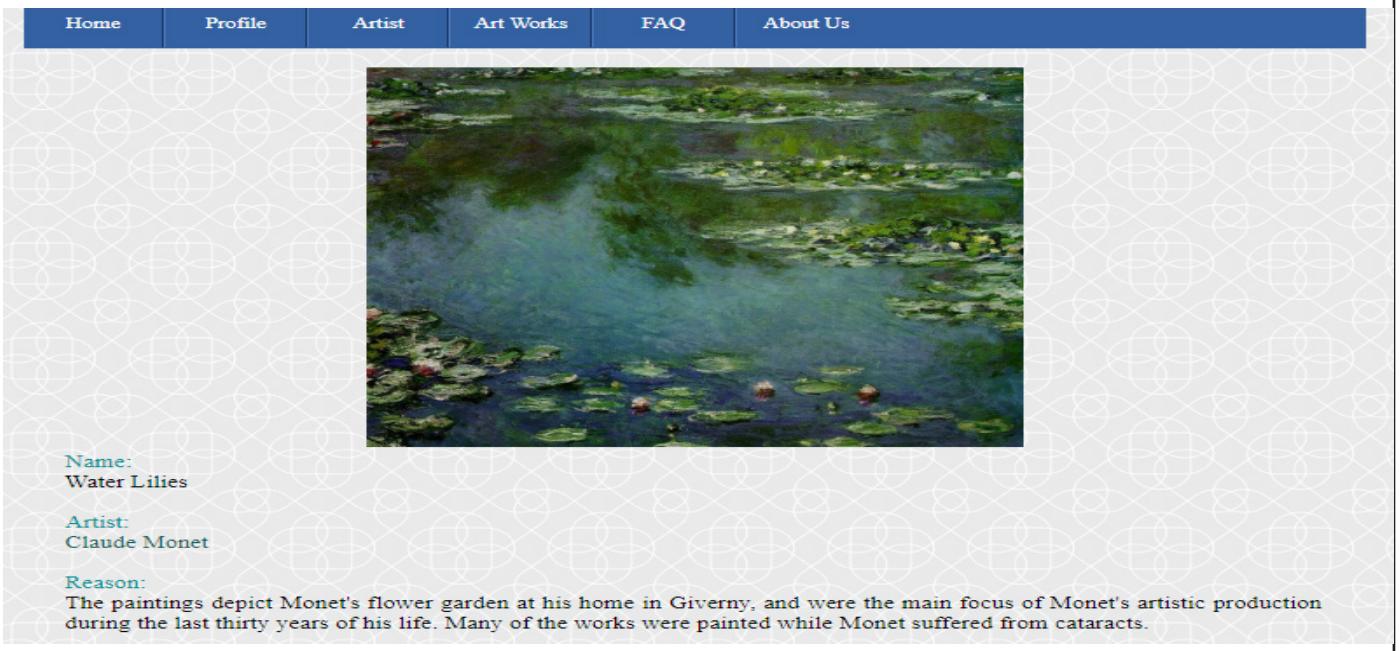


Fig-11 : Art Page

Giving review on an art work :

To give review on any art work we have at first create an account. Then we need to log in. After that we can give comment. We can see others comment also. We can only edit or delete our own comment but not others.

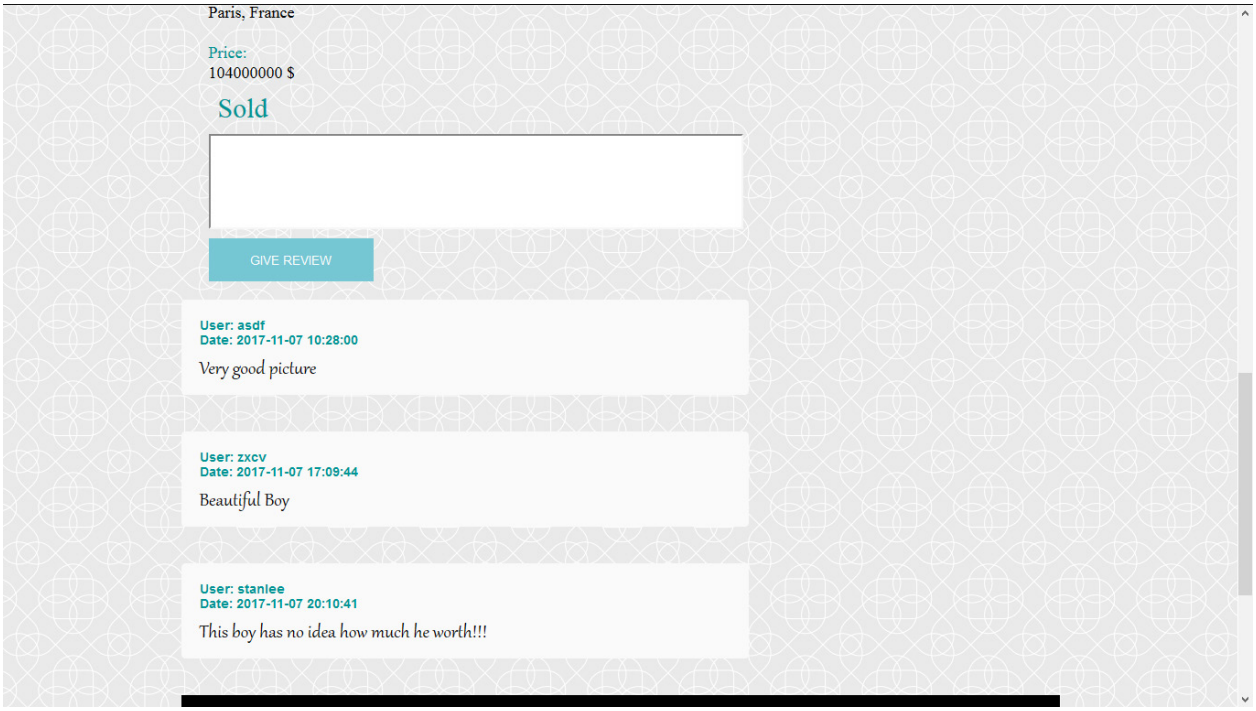
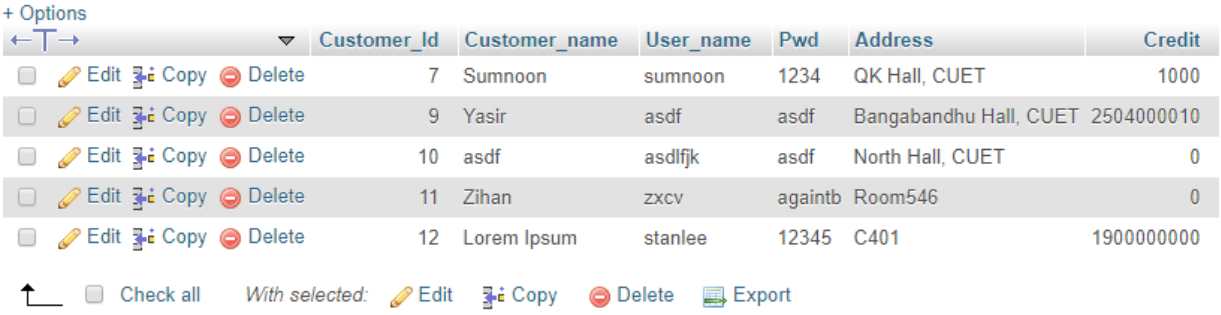


Fig-12 : Review

Buying an art work :

If we want to buy any art work, at first we have to be logged in our account.Then we can see the price of a certain art work and if we like the art work we can click buy button then trainsection page will show.After that we have to insert the phone number that has been used to send money also the bikash transection id. After that credit will be shown on our profile page.



+ Options

	Customer_Id	Customer_name	User_name	Pwd	Address	Credit
Edit Copy Delete	7	Sumnoon	sumnoon	1234	QK Hall, CUET	1000
Edit Copy Delete	9	Yasir	asdf	asdf	Bangabandhu Hall, CUET	2504000010
Edit Copy Delete	10	asdf	asdlfjk	asdf	North Hall, CUET	0
Edit Copy Delete	11	Zihan	zxcv	againtb	Room546	0
Edit Copy Delete	12	Lorem Ipsum	stanlee	12345	C401	1900000000

Check all *With selected:* Edit Copy Delete Export

Fig-13 : Customer table before updating credit

Fig-14 : Buying an art

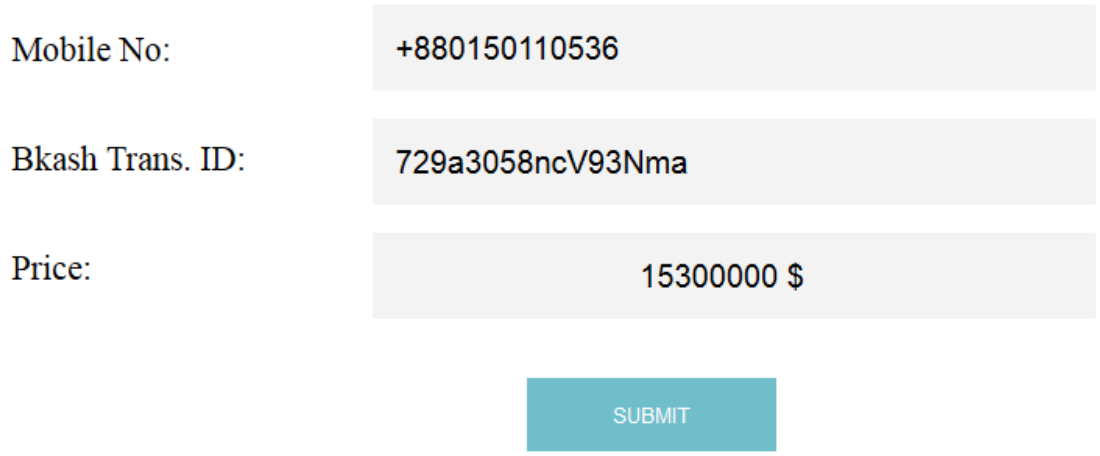


Fig-15: Transection Page

$sql = “SELECT * FROM customer WHERE User_name=’$cname’”;

$credit = $row[‘Credit’] + $price;

$sql = “UPDATE customer SET Credit = ‘$credit’

WHERE User_name = ‘$cname’ “;

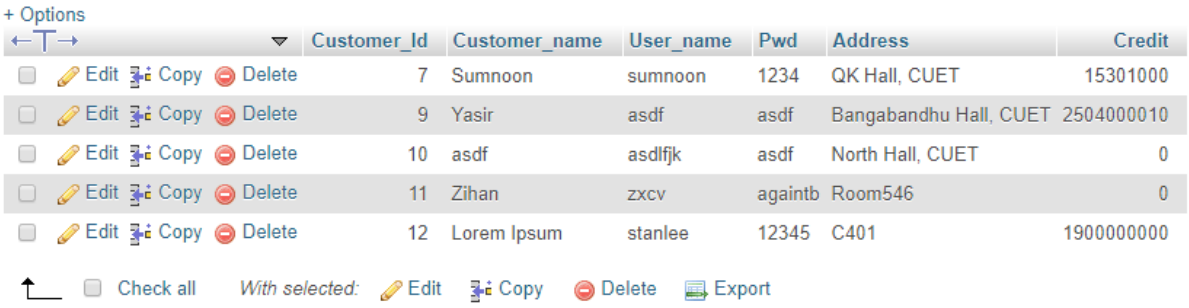


Fig-16 : Customer Table after buying

Using these three statement we will update our user profile.After choosing to buy an art work, at first we will select current credit of our user profile than we will add price of the art work with it and update the credit. Then the page of art work will not show any buy button instead it will show ‘SOLD’.

Fig-17 : Art has been sold

$cid = $row[‘Customer_Id’];

$sql = “UPDATE art SET Cid = ‘$cid’ WHERE Art_Id = ‘$aid’ “;

By aforementioned two statement at first we have taken the customer id than we will set the Cid on Art Table which was 0 before the transection.If Cid is 0 or Null that means this art has not been bought by any one. But other than 0 it can be will certify that the art has been bought by user who has user id equals to Cid.

Chapter 6

Conclusion

Limitation:

We have some drawbacks in our webisite, here we try to figure out Major draw-back of our website.

- This website is only used for educational purpose not intended for commarcial interest.
- Here we can not search any specific artwork through this website.
- Security of our website is not fully secured. We are trying to enhance security level of this website.
- Here customer can not vote favorite for any artwork.
- Customers can not pay via visa,master card. They can only pay via bikash. Also there is no way to confirm that the given transection id of bikash payment is valid as we are using a localhosting server. If we can use a server it could be easier.
- The collection of artwork is not vast.Also we are using only artwork which are virtually just some wallpapers.
- In the website we can not see all artwork of a single artist in single page.
- There is no email verification system for user accounts,so we can not verify nor send any important mail to our user.
- We could not handle the foreign key properly.

Future Recommendation:

We have already set some future goals how our website will be updated for giving our users better and more advanced facilities. Besides, we will also try to reduce the limitation of current website.Some future aspect that will be taken is given below.

- We will try to include search option so that user can search his/her desired artworks easily.
- Now our website is vulnerable to any cyber terrorism. We will to try to increase our security so that it can defend some basic attack.
- We will secure our transection process to defend against any scamming. Also we will improve the transection process to receive money not only from bikash but also from Visa, MasterCard etc.
- We will enrich our website by including more arts from different artist also we will add more category of artwork like calligraphy, sculpture, photography etc
- Now this website is just host oriented. User can not upload any artworks.We will try to create the website flexible so that user can upload any artwork.
- We will try to improve the website so that it becomes more user friendly.Also user can see all artworks of an artist in one page.
- There is a review system in our website for every artwork.In future we will try to include a suggestion box so that user can give their useful opinion to improve our website also they can review our website.
- There will be mailing system so the account can verified and we can send important mail to our user.

Discussion :

It has been a matter of immense pleasure, honor and challenge to have this opportunity to take up this project. In this era of information and technology, everyone should have access to information and right to utilize them.

We think that this website will help its user in this respect. It will make easier for admin to manage traffiking activities through this website. This website will help the students to knowledge about various category of artworks. Also there is a section for every arts and their artist which will be helpful for learning purpose. It is helpful for people who does not have the luxery to lose time for going and searching for art of his/her favorite artist. Also we will include a suggestion box so that user can send their valuable idea or suggestion for future improvement.

Finally, we hope that this website will help to make amicable relationship among the artworld community.

REFERENCES

:

- https://www.w3schools.com/html/default.asp
- https://www.w3schools.com/css/default.asp
- https://www.w3schools.com/sql/default.asp
- https://www.w3schools.com/php/default.asp