FLAT MANAGEMENT SYSTEM

Database Project Report

Group-11

Report submitted April 03, 2022

A project submitted to Dr. Rudra Pratap Deb Nath, Associate Professor, Department of Computer Science and Engineering, Chittagong University (CU) in partial fulfillment of the requirements for the Database Systems Lab course. The project is not submitted to any other organization at the same time.

Table 1: Details of Group-11

Roll Id	Name	Sigature	Date	Supervisor Approval
				keep it blank
19701075	Talhabin Siam	Md Siam	03-04-2022	
19701044	Jahangir Alam Jehad	Jehad	03-04-2022	
19701074	Abu Noman Shawn Sikder	Shawn	03-04-2022	
19701071	Rabbi Hasan	Rabbi	03-04-2022	
19701059	Md Hasan Mia	Hasan	03-04-2022	

Contents

List of Figures

List of Tables

Listings

Abstract

We chose the flat management system. Many of renters are facing problems when searching a flat, cottage or mess. Also flat owner can easily handle their flats, cottages. People can easily see the flats and find their expected flats. Also mess manager can add room when there is a vacancy in his/her mess. For solving this system we chose the flat management system.

1 Introduction

We are exercising this project for learning easy management of flat. The objective of this course is to develop a database application system by applying the theories, methodologies, tools, and technologies we are learned in Database System[CSE-413].

1.1 Background and Motivation

This project is highly important for us to overcome the limitations of the manual data management system. On account of this problem many companies organizations lose their precious time and money. Currently it is really difficult to manage a large amount of data and search a particular data. Our project can solve the limitations easily. Because of this process time will be saved and so money will be saved.

1.2 Problem Statement

The Flat Management system is developed for making it easy to find the cottage/mess/flat. This system will contain all information about cottages/messes/flats from where customers can find out their preferable cottages/messes/flats according to their budgets/ choices. Users can perform different kinds of queries based on their location preferences, budget etc.

1.3 System Definition

Flat Management system is used for managing the cottage/mess/flat virtually which can store information about cottages/messes/flats in where customers can easily find out their expected cottages/messes/flats.

A system definition example of a Conference planning system

"A computerized system used to control the ICCIT conference by registering participants and their payments to organizers using invoicing and other reporting methods. Controlling should be easy to learn, as ICCIT conferences use unpaid and untrained labor."

1.4 System Development Process

We are following several steps. At first, we collect requirements. Then we analyze requirements and find some entity types and attributes. With those entity types and attributes we form an Entity Relationship(ER) diagram. After that we map it onto a relational schema. Then we normalize our database. For developing this system we use PHP as backend language and javascript, HTML, CSS, Jquery for designing frontend.

1.5 Organization

Section 1 gives the overview of the project, Section 2 describes how the project and the resources are managed. Section 3 describes how requirements are gathered and analysed. Section 4 describes how we model our database using Entity Relationship(ER) model, Section 5 describes how we convert our Entity Relationship(ER) model into Relational model, Section 6 describes functional dependencies of each relation schema from previous, Section 5 and shows that they are normalized up to 3NF or BCNF, Section 7 describes the overall architecture of our database system, Section 8 describes the whole process of front end and back end of our database system, Section 9 describes the success of our product and manual of the system for users, Section 10 describes the formal proof and disclosure of Flat Management System, Section 11 describes how to install and configure our system so that a non-technical user can use our system, Finally, the conclusion and the pointers to future work is outlined in section 12.

2 Project Management

Our team consists of 5 members. On completion of different virtual meeting we selected Flat Management System as our database management system project.
Project Leader : Md Siam
Requirement Analysis :

1. Md Siam
2. Rabbi Hasan
3. Abu Noman Shawn Sikdar
4. Jahangir Alam Jehad
5. Hasan Mia

ER Diagram

1. Md Siam
2. Rabbi Hasan
3. Abu Noman Shawn Sikdar

Logical Modeling

1. Md Siam
2. Rabbi Hasan
3. Md Hasan Mia

Normalization

1. Rabbi Hasan
2. Md Siam
3. Jahangir Alam Jehad
4. Md Hasan Mia
5. Abu Noman Shawn Sikdar

System deployment : Database setup in pc

1. Md Siam
2. Jahangir Alam Jehad

Backend:

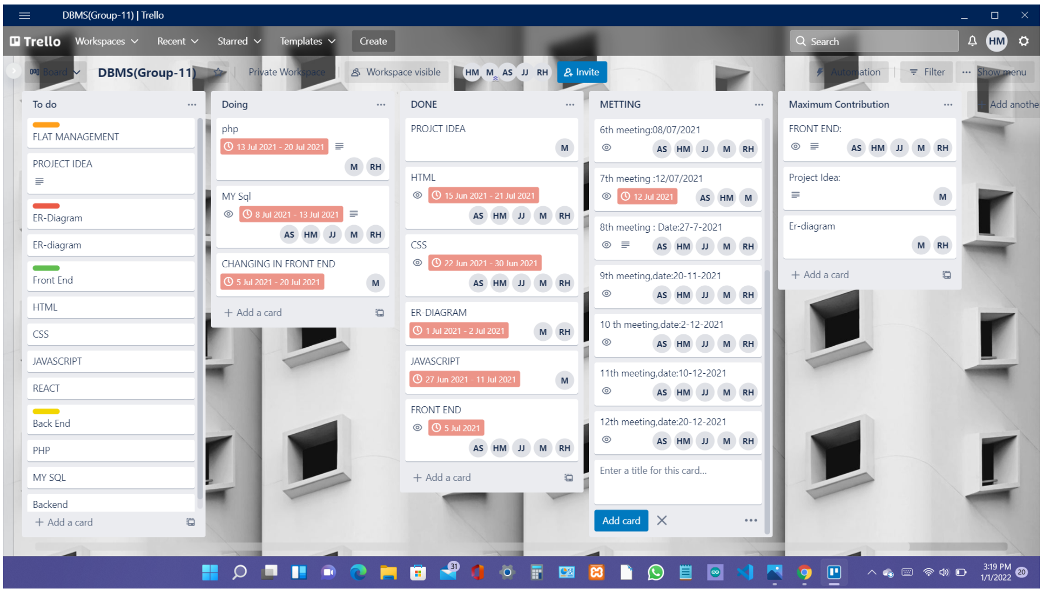
1. Md Siam
2. Md Hasan

Database:

1. Md Siam
2. Rabbi Hasan
3. Abu Noman Shawn Sikdar
4. Jahangir Alam Jehad
5. Hasan Mia

Front-end :

1. Jahangir Alam Jehad
2. Abu Noman Shawn Sikdar
3. Md Hasan Mia
4. Md Siam
5. Rabbi Hasan



To do

- FLAT MANAGEMENT
- PROJECT IDEA
- ER-Diagram
- ER-diagram
- Front End
- HTML
- CSS
- JAVASCRIPT
- REACT
- Back End
- PHP
- MY SQL
- Backend

+ Add a card

Doing

- php — 13 Jul 2021 - 20 Jul 2021 — M RH
- MY Sql — 8 Jul 2021 - 13 Jul 2021 — AS HM JJ M RH
- CHANGING IN FRONT END — 5 Jul 2021 - 20 Jul 2021 — M

+ Add a card

DONE

- PROJCT IDEA — M
- HTML — 15 Jun 2021 - 21 Jul 2021 — AS HM JJ M RH
- CSS — 22 Jun 2021 - 30 Jun 2021 — AS HM JJ M RH
- ER-DIAGRAM — 1 Jul 2021 - 2 Jul 2021 — M RH
- JAVASCRIPT — 27 Jun 2021 - 11 Jul 2021 — M
- FRONT END — 5 Jul 2021 — AS HM JJ M RH

+ Add a card

METTING

- 6th meeting:08/07/2021 — AS HM JJ M RH
- 7th meeting :12/07/2021 — 12 Jul 2021 — AS HM M
- 8th meeting : Date:27-7-2021 — AS HM JJ M RH
- 9th meeting,date:20-11-2021 — AS HM JJ M RH
- 10 th meeting,date:2-12-2021 — AS HM JJ M RH
- 11th meeting,date:10-12-2021 — AS HM JJ M RH
- 12th meeting,date:20-12-2021 — AS HM JJ M RH

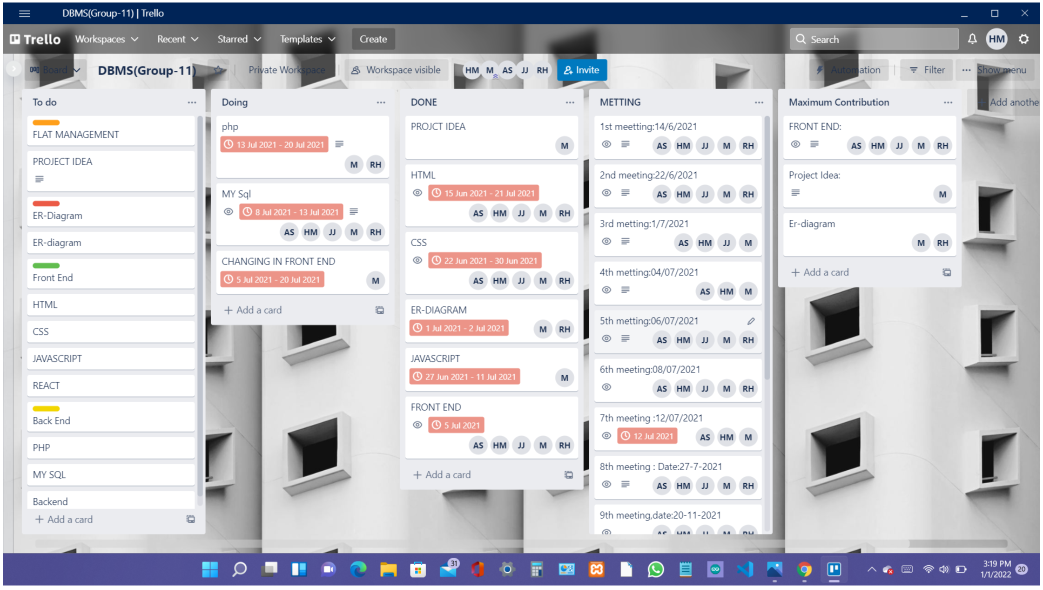
Enter a title for this card...

Add card

Maximum Contribution

- FRONT END: — AS HM JJ M RH
- Project Idea: — M
- Er-diagram — M RH

+ Add a card



To do

- FLAT MANAGEMENT
- PROJECT IDEA
- ER-Diagram
- ER-diagram
- Front End
- HTML
- CSS
- JAVASCRIPT
- REACT
- Back End
- PHP
- MY SQL
- Backend

+ Add a card

Doing

- php — 13 Jul 2021 - 20 Jul 2021 — M RH
- MY Sql — 8 Jul 2021 - 13 Jul 2021 — AS HM JJ M RH
- CHANGING IN FRONT END — 5 Jul 2021 - 20 Jul 2021 — M

+ Add a card

DONE

- PROJCT IDEA — M
- HTML — 15 Jun 2021 - 21 Jul 2021 — AS HM JJ M RH
- CSS — 22 Jun 2021 - 30 Jun 2021 — AS HM JJ M RH
- ER-DIAGRAM — 1 Jul 2021 - 2 Jul 2021 — M RH
- JAVASCRIPT — 27 Jun 2021 - 11 Jul 2021 — M
- FRONT END — 5 Jul 2021 — AS HM JJ M RH

+ Add a card

METTING

- 1st meetting:14/6/2021 — AS HM JJ M RH
- 2nd meeting:22/6/2021 — AS HM JJ M RH
- 3rd metting:1/7/2021 — AS HM JJ M
- 4th metting:04/07/2021 — AS HM M
- 5th metting:06/07/2021 — AS HM JJ M RH
- 6th meeting:08/07/2021 — AS HM JJ M RH
- 7th meeting :12/07/2021 — 12 Jul 2021 — AS HM M
- 8th meeting : Date:27-7-2021 — AS HM JJ M RH
- 9th meeting,date:20-11-2021

Maximum Contribution

- FRONT END: — AS HM JJ M RH
- Project Idea: — M
- Er-diagram — M RH

+ Add a card

Figure: Group meeting and schedule using ”Trello” software.

3 Requirement Gathering and analysis

University Students are our stackholders who are sharing their problems when they are searching the flat, cottage or mess for renting. Our team members are talking face to face with different university students to gather information for our project requirement.
Requirement :

1. Flat owner section
2. Renter/general user section
3. Admin section
4. Renter can search flats with a filter.
5. Renter can request for booking flats.
6. Admin can add, remove and update user.
7. Admin can see all users and flats.
8. Flat owners can add ,remove and update flats.
9. Flat owners can see all requests for a flat.
10. Flat owners can see the booked and unbooked users.

System Requirement :

1. HTML
2. CSS
3. BOOTSTRAPhttps
4. JAVASCRIPT
5. JQUERY
6. Mysql
7. XAMPP(Apache)

4 Conceptual Modelling

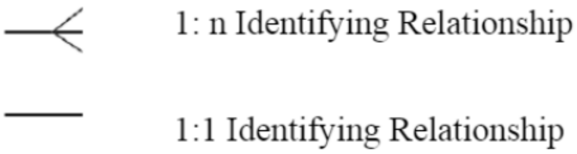
ER Diagram stands for Entity Relationship Diagram, also known as **ERD** is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

1. Entity type: A group of definable things, such as User, whereas the entity would be the specific User.

2. Entity set: Same as an entity type,but defined at a particular point in time, such as User registration first day of the month. Relationship: How entities act upon each other or are associated with each other.

3. Attribute: A property or characteristic of an entity. Often shown as an oval or circle.

4. Cardinality: Defines the numerical attributes of the relationship between two entities or entity sets . The three main cardinal relationship are (1).one-to-one,(2). one-to-many and (3).many-to-many.



According to requirements, we have to store about the admin, flat-owner, flat, facilities and renter. Finally, we find here 5 entity types like General User, Flat Owner, Admin, Flat, Facilities.

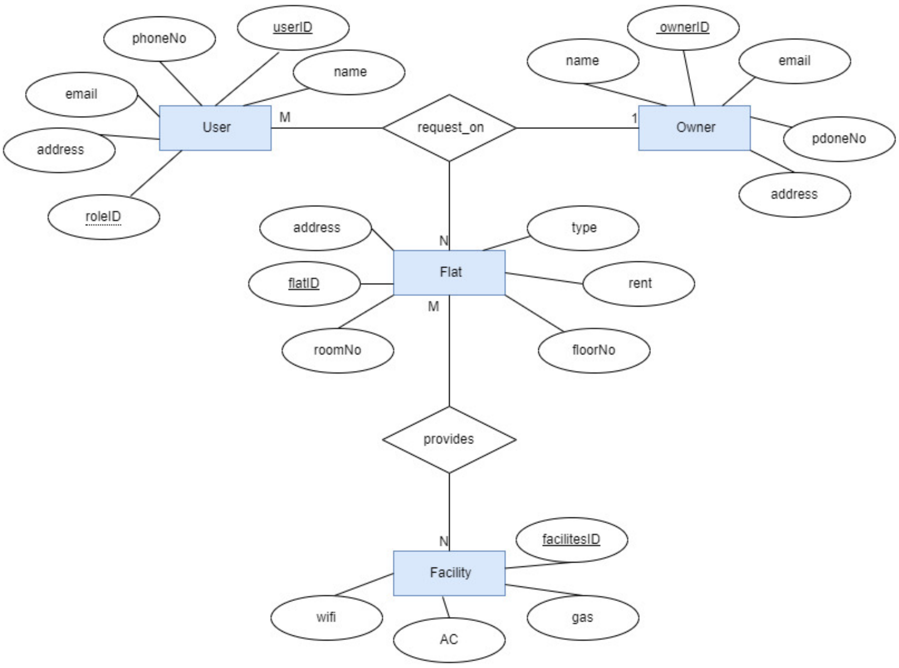


Figure-1: Entity Relationship(ER) Diagram.

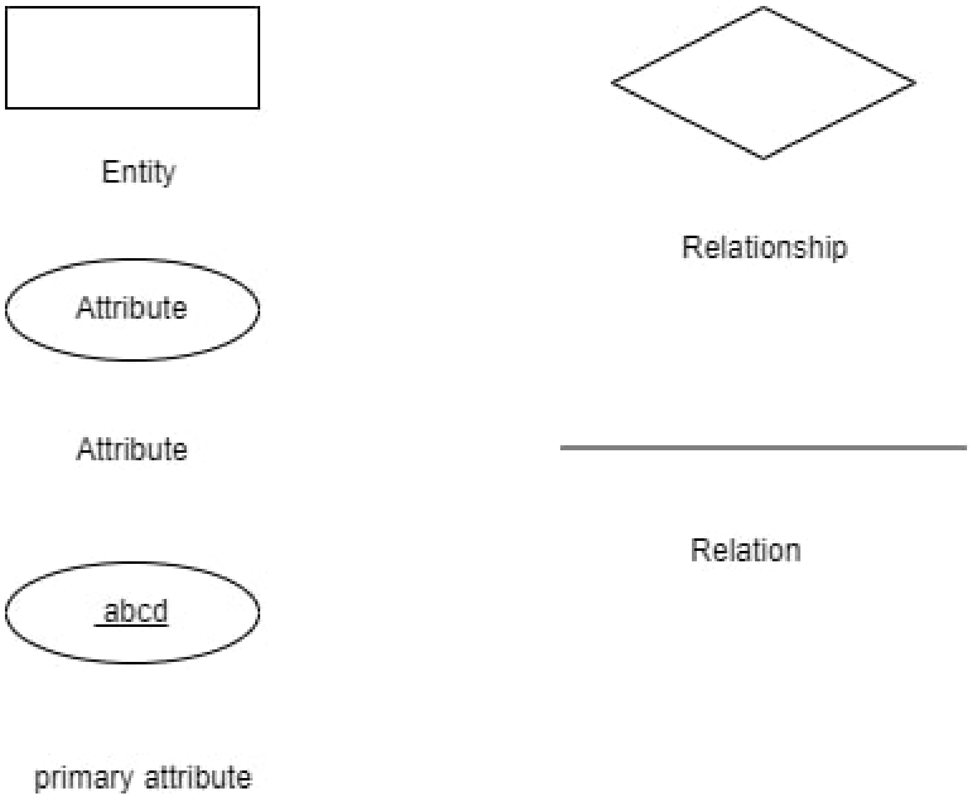


Figure-2: Legends used for ER diagram.

Unary relationship (recursive): A unary relationship, also called recursive, is one in which a relationship exists between occurrences of the same entity set. In this relationship, the primary and foreign keys are the same, but they represent two entities with different roles.

Binary relationship: When there are exactly two entity sets participating in a relationship then such type of relationship is called binary relationship.

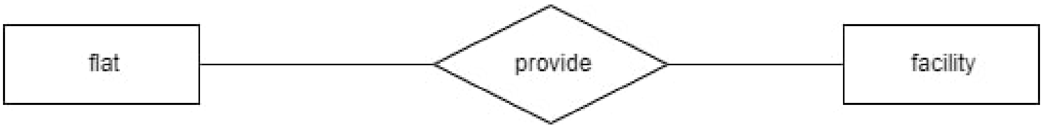


Figure-2.1: Binary Relationship.

Ternary relationship: A ternary relationship is an association among three entities. This type of relationship is required when binary relationships are not sufficient to accurately describe the semantics of the association. The ternary relationship construction is a single diamond connected to three entities as shown in Figure2.2.

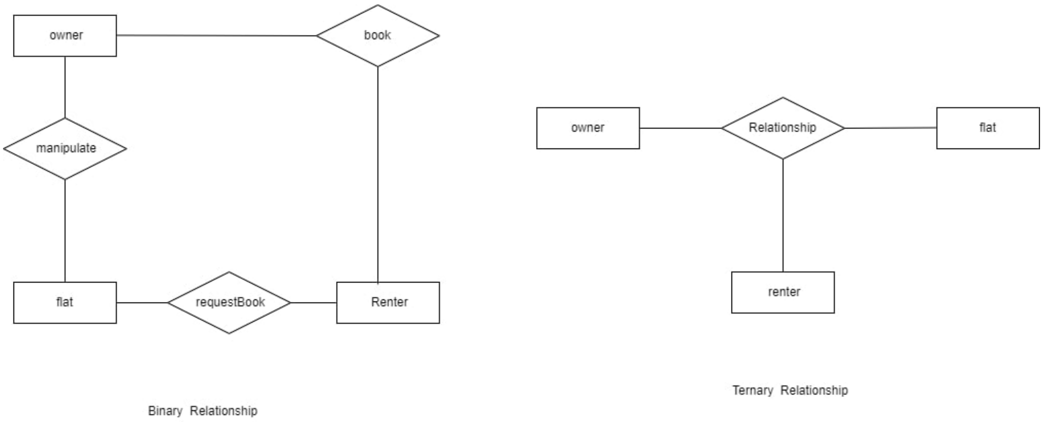


Figure-2.2: Ternary Relationship

5 Logical Modelling

A logical data model establishes the structure of data elements and the relationships among them. The relational data model provides a standard way of representing and querying data that can be used by any application. At the beginning, developers can recognize that the main strength of the relational database model is in its use of tables, which is an intuitive, efficient and flexible way to store and access structured information.
This logical model captured from *phpMyadmin* which is formed by our established database:

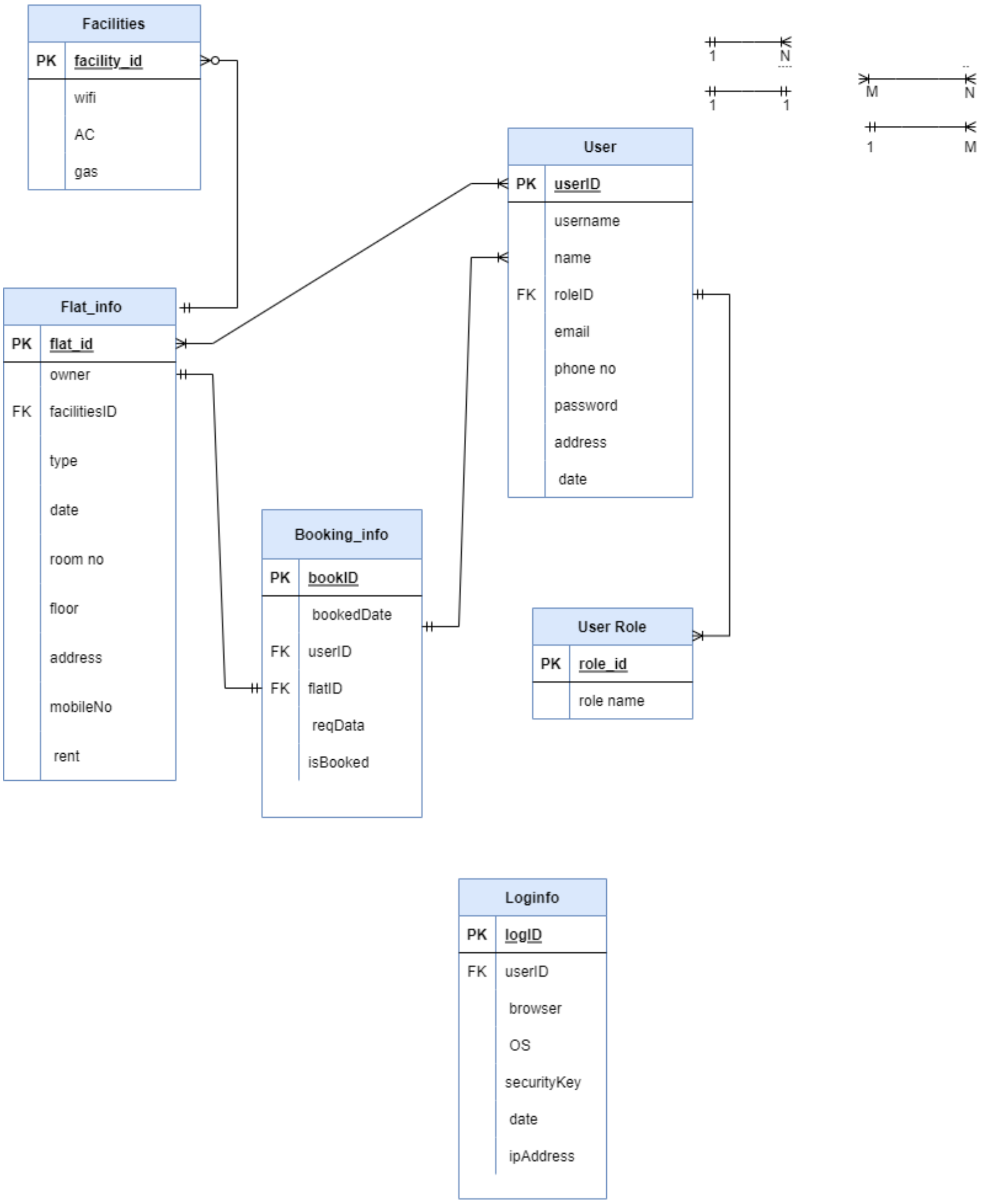


Figure-3: Relational modeling

6 Normalization

Normalization is a process of avoiding as possible redundancy from a relation or a set of relational table . Redundancy may occur many problems like

1. Insertion anomalies
2. Deletion anomalies and
3. Updating anomalies.

Insertion anomalies : An insertion anomaly is the unability to add data to the database due to the absence of other data . Suppose,a new Flat-ID is inserted into flat-Info table but won't insert data under the tuple then the data is made inconsistencies due to omission in database.

Deletion anomalies : A deletion anomaly occurs when you delete a record that may contain attributes that shouldn't be deleted. For instance,if we delete an record in an attributes , we will have a chance to loss other attributes information in a corresponding table .
Updating anomalies : An update anomaly is a data inconsistency that results from data redundancy and a partial update .
So,normalization helps to eliminate this type problem and makes it easier to retrieve data.

6.1 First Normal Form(1NF)

"A relation does not contain any multi-valued attributes and composite attributes." In first normalization,

1. There are only Single Valued Attributes.
2. Attribute Domain does not change
3. There is a unique name for every Attribute/Column.
4. The order in which data is stored does not matter.

Multi-valued attributes : A multi-valued attribute of an entity is an attribute that can have more than one value associated with the key of the entity. For example , a large company can have many divisions, some of them possibly in different cities.
Composite attributes : An attribute composed of many other attributes is called as composite attribute.

6.2 Second Normal Form(2NF)

Second Normal Form (2NF) is based on the concept of full functional dependency . Second Normal Form applies to relations with composite keys , that is, relations with a primary key composed of two or more attributes . A relation with a single-attribute primary key is automatically in at least 2NF . In second normal form

1. It must be 1st normal form and
2. There must be no partial dependency .

Partial Dependency : If the proper subset of candidate key determines non-prime attribute, it is called partial dependency[1].

6.3 Third Normal Form(3NF)

A relation is in third normal form , if there is no transitive dependency for non-prime attributes as well as it is in second normal form . A relation is in 3NF[2]. if at least one of the following A→B:

1. It must be 2NF.
2. There must not be any transitivity and
3. A→B is trivial where $A \subset RandB \subset R$.

Transitivity : If A→B and B→C are two functional dependencies ,then A→C is called transitive dependency/transitivity .

6.4 Boyce-Codd Normal Form(BCNF) :

Boyce–Codd Normal Form (BCNF) is based on functional dependencies that take into account all candidate keys in a relation; however, BCNF also has additional constraints compared with the general definition of 3NF.
In BCNF:

1. It must be 3NF .
2. A→B is trivial where$A \subset RandB \subset R$. and
3. There must be a super key(SK).

[1] src-geekforgeek.com/....

[2] 3NF:For each FDs,LHS must be CK/SK OR RHS is a prime attributes

7 Normalization: in the context of project

KEYS : Keys in DBMS is an attribute or set of attributes which helps to identify a row(tuple) in a relation . They allow to find the relation between two tables . Keys help uniquely identify a row in a table by a combination of one or more columns in that table . Key is also helpful for finding unique record or row from the table . Database key is also helpful for finding unique record or row from the table.

1. **Super Key :** A super key is a group of single or multiple keys which identifies rows in a table.

2. **Primary Key :** A Primary key is a column or group of columns in a table that uniquely identify every row in that table.

3. **Candidate Key :** A Candidate key is a set of attributes that uniquely identify tuples in a table. Candidate Key is a super key with no repeated attributes .

4. **Alternate Key :** An Alternative key is a column or group of columns in a table that uniquely identify every row in that table.

5. **Foreign Key :** A Foreign Key is a column that creates a relationship between two tables. The purpose of Foreign keys is to maintain data integrity and allow navigation between two different instances of an entity .

6. **Compound Key :** A Compound key has two or more attributes that allow you to uniquely recognize a specific record. It is possible that each column may not be unique by itself within the database .

7. **Prime Attribute :** A prime attribute of the database tables which are candidate keys of the database tables are called prime attributes.

8. **Functional Dependency :** A functional dependency is a constraint that specifies the relationship between two sets of attributes where one set can accurately determine the value of other sets. It is denoted as X → Y, where X is a set of attributes that is capable of determining the value of Y. The attribute set on the left side of the arrow, X is called Determinant, while on the right side, Y is called the Dependent.

Table:User/SignUp
User{ userID , email, username , phoneNo , name , password , date , roleID , address. }

Functional Dependency :F^+ clouser

1. {userID}→{username , email , phoneNo , name , date , address.}

Candidate Key : userID..
Prime Attribute : userID.
Non prime Attribute:email,phoneNO username , name , password , date , roleID , address.
Foreign Key : roleID.

1. No multi-valued/composite attributes,So it's 1NF.
2. There is no partial dependency,So it's 2NF.
3. Here,the right side attributes userID is super keys and there is no transitivity/transitive dependency,So it's 3NF.

Table : FlatInfo
FlatInfo{flatID , owner , type , floor , room , facilitiesID ,address}

Functional Dependency :F^+ clouser

1. {flatID}→{owner , type , floor , room , facilitiesID , address}

Candidate Key : flatID
Prime Attribute : flatID
Non prime Attribute : owner , type , floor , room , facilitiesID , address
Foreign Key : facilitiesID

1. No multi-valued/composite attributes,So it's 1NF.
2. Single Primary Key and no partial dependency,So it's 2NF.
3. There is no transitivity,So it's 3NF.

Table:LogInfo
LogIn{logID , userID ,ipAddress , browser , OS , password , data}

Functional Dependency : F^+ clouser

1. {logID}→{userID , ipAddress , browser , OS , password , data}

Candidate Key: logID
Prime Attribute : logID
Non Prime Attribute : userID , ipAddress , browser , OS , password , data
Foreign Key:userID

1. No multivalued/composite attributes,So it's 1NF.
2. Single Primary Key and no partial dependency,So it's 2NF.
3. There is no transitive dependency,So it's 3NF.

Table:UserRole
UserRole(roleID,role_name)

Functional Dependency : F^+ clouser

1. {roleID}→{role_name}

Primary Key : roleID
Prime Attribute : roleID
Non Prime Attribute : role_name

1. No multivalued/composite attributes,So it's 1NF.
2. Single Primary Key and no partial dependency,So it's 2NF.
3. There is no transitive dependency,So it's 3NF.

Table:Facility
Facility{facilitiesID,wifi,AC,gas}

Functional Dependency : F^+ clouser

1. {facilitiesID}→{wifi , AC , gas}

Candidate Key/Primary Key : facilitiesID
Prime Attribute : facilitiesID
Non Prime Attribute : wifi , AC , gas

1. No multivalued/composite attributes,So it's 1NF.
2. Single Primary Key and no partial dependency,So it's 2NF.
3. There is no transitive dependency,So it's 3NF.

Table : BookingInfo
BookingInfo{BookID , userID , bookDate , flatID , isBooked , reqDate}

Functional Dependency:F^+ clouser

1. {bookID}→{userID , bookDate , flatID , isBooked ,reqDate}

Candidate Key : BookID
Foreign Key/Primary Key: flatID
Prime Attribute: BookID
Non Prime Attribute: userID , bookDate , flatID , isBooked , reqDate.

1. No multi-valued/composite attributes,So it's 1NF

2. Single Primary Key and no partial dependency,So it's 2NF.

3. There is no transitive dependency,So it's 3NF.

8 System Architecture

admin : Admin is the controller of this system.
Home : Home page is landing page of our developed system
Dashboard : When user login into website see his page
user database: User database contains all users o information type
Flat database: Flat database contain all information of flats.

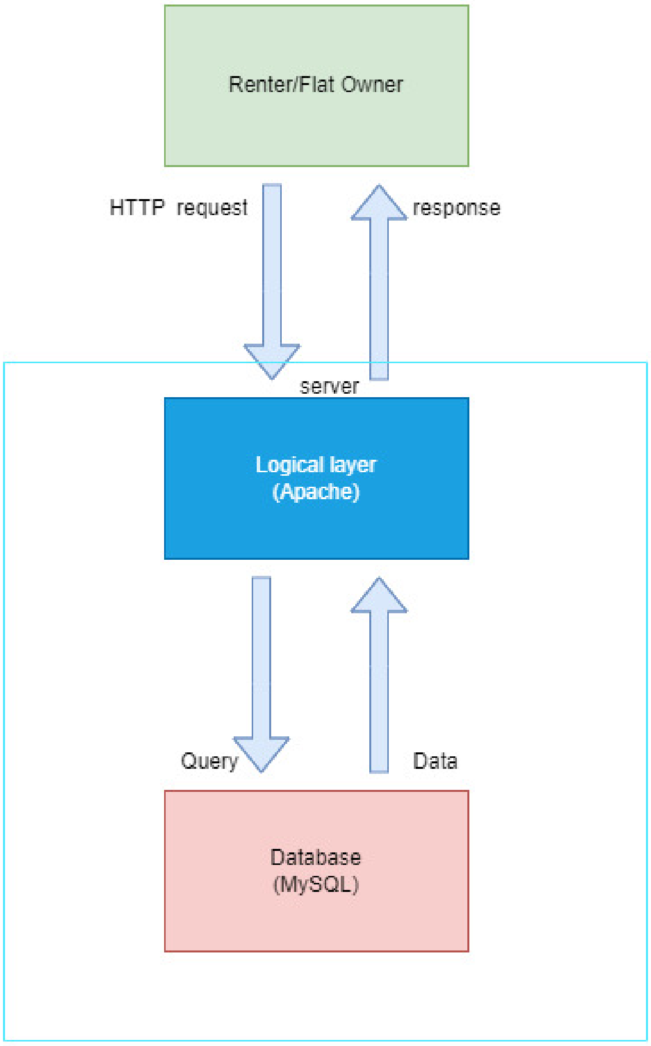


Figure-4: System Architecture.

9 Implementation

Here given some Data Definition Language(DDL) query which are used in our system
Listing 14

```
CREATE TABLE users (
  ID int(16) UNSIGNED NOT NULL,
  username varchar(60) COLLATE utf8_unicode_ci NOT NULL,
  email varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  password varchar(60) COLLATE utf8_unicode_ci NOT NULL,
  role varchar(60) COLLATE utf8_unicode_ci NOT NULL,
  name varchar(60) COLLATE utf8_unicode_ci NOT NULL,
  address varchar(250) COLLATE utf8_unicode_ci NOT NULL,
  phone varchar(14) COLLATE utf8_unicode_ci NOT NULL,
  date datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

Listing 1: A SQL command for Creating table named USERS

```
 ALTER TABLE flat_info
  ADD PRIMARY KEY (ID);
```

Listing 2: A SQL command for setting primary key.

```
ALTER TABLE users
  ADD PRIMARY KEY (ID);
```

Listing 3: A SQL command for setting primary key.

```
CREATE TABLE bookinginfo (
  bookID int(16) UNSIGNED NOT NULL,
  userID int(16) UNSIGNED NOT NULL,
  flatID int(16) UNSIGNED NOT NULL,
  date datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

Listing 4: A SQL command for Creating table named booking-info

```sql
CREATE TABLE flatinfo (
  flatID int(16) UNSIGNED NOT NULL,
  userID int(16) UNSIGNED NOT NULL,
  status enum('Available','Booked','Trash') COLLATE
     utf8_unicode_ci DEFAULT 'Available',
  type enum('Flat','Mess','Cottage') COLLATE
     utf8_unicode_ci DEFAULT 'Flat',
  floor varchar(10) COLLATE utf8_unicode_ci NOT NULL
     ,
  rent int(10) NOT NULL,
  room varchar(20) COLLATE utf8_unicode_ci NOT NULL,
  address varchar(250) COLLATE utf8_unicode_ci NOT
     NULL,
  mobile varchar(15) COLLATE utf8_unicode_ci NOT
     NULL,
  facID int(16) UNSIGNED NOT NULL,
  date datetime NOT NULL,
  images longtext COLLATE utf8_unicode_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=
   utf8_unicode_ci;
```

Listing 5: A SQL command for Creating table named Flat-info

```sql
 CREATE TABLE loginfo (
  logID int(16) UNSIGNED NOT NULL,
  userID int(16) UNSIGNED NOT NULL,
  ipAddress varchar(15) COLLATE utf8_unicode_ci
     DEFAULT NULL,
  os varchar(10) COLLATE utf8_unicode_ci DEFAULT
     NULL,
  browser varchar(20) COLLATE utf8_unicode_ci
     DEFAULT NULL,
  date datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=
   utf8_unicode_ci;
```

Listing 6: A SQL command for Creating table named logInfo

```sql
SELECT * FROM user_role WHERE roleID = '$role_Id';
```

Listing 7: A SQL command for Creating table named user-role

```
SELECT user.*, loginfo.logID, loginfo.userID FROM
    loginfo    RIGHT JOIN user on loginfo.userID =
    user.userID WHERE       loginfo.logID = '$logID'
     AND loginfo.securityKey='$logKey;
```

Listing 8: A SQL command for connecting logInfo and user

```
SELECT flatinfo.*, facilities.*, facilities.facID
    AS facID   FROM flatinfo LEFT JOIN facilities ON
     flatinfo.facID =      facilities.facID ORDER BY
     date ASC;
```

Listing 9: A SQL command for connecting flatinfo and facilities

```
DELETE FROM flatinfo WHERE userID='$user_id' AND
    flatID='$id';
```

Listing 10: A SQL command for deleting flatinfo

```
UPDATE flatinfo SET status='Booked', bookedUser='
    $user_i' WHERE flatID='$id' AND userID='$user_id
    ';
```

Listing 11: A SQL command for Setting status flatinfo

```
UPDATE flatinfo SET status='Available', bookedUser=
    NULL      WHERE flatID='$id' AND userID='
    $user_id';
```

Listing 12: A SQL command for updating status flatinfo

```
SELECT * FROM user WHERE username='$username';
```

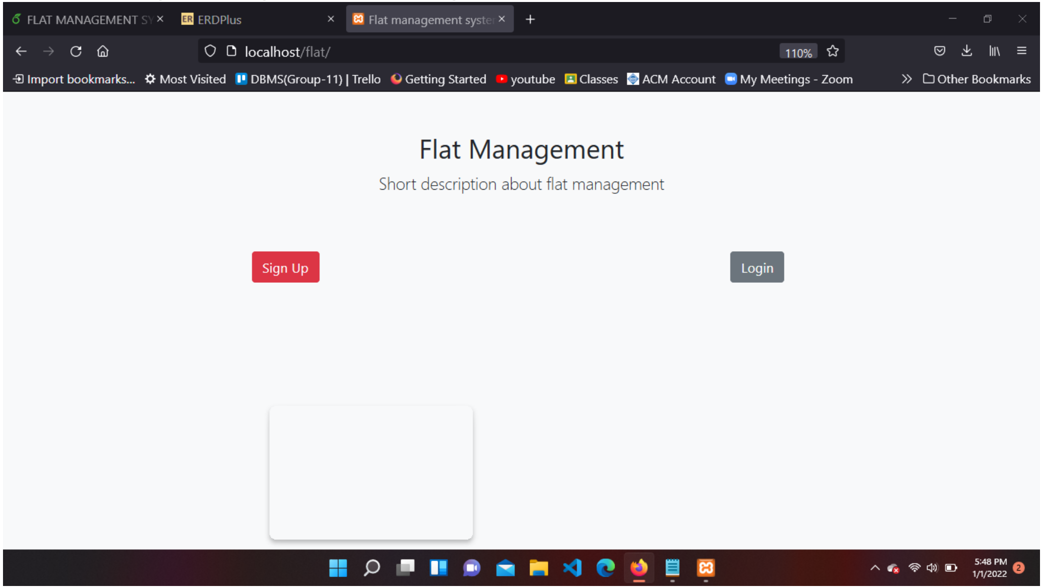
Listing 13: A SQL command for finding username from user table

```
UPDATE user SET password='$password', email='$email
    ',roleID='$type', name='$name', address='
    $address',phone='$phone' WHERE userID='$uid';
```

Listing 14: A SQL command for update user

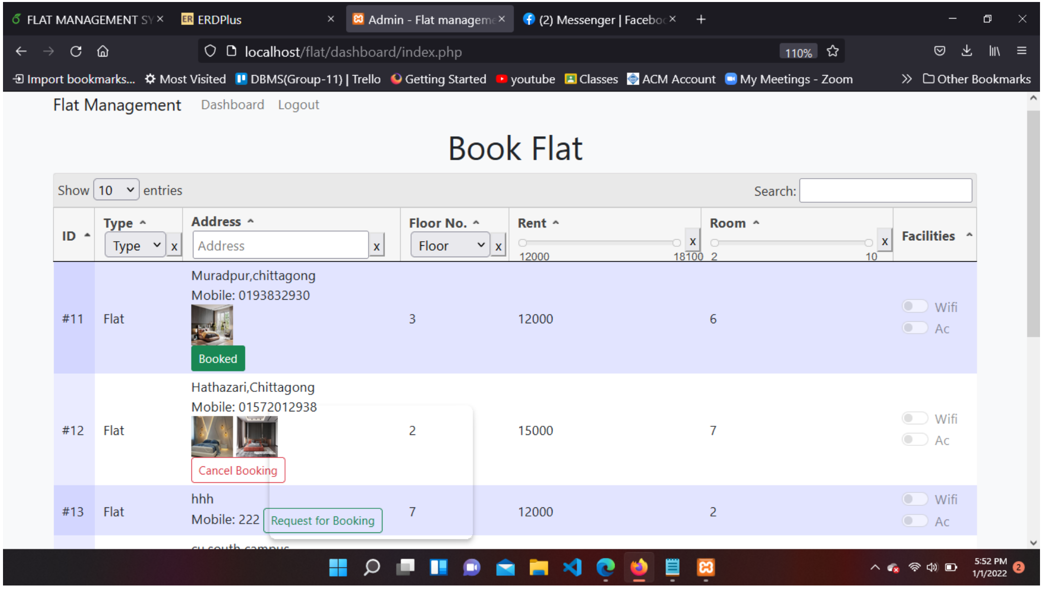
10 Validation

A simple user manual: When the user starts the application, he/she will find the landing page. A snap of lading page is given below. From this page, a user can go to login form or registration form.


FLAT MANAGEMENT SY
ERDPlus
Flat management syste
localhost/flat/
110%
Import bookmarks... Most Visited DBMS(Group-11) | Trello Getting Started youtube Classes ACM Account My Meetings - Zoom Other Bookmarks
Sign Up
Login
5:48 PM
1/1/2022

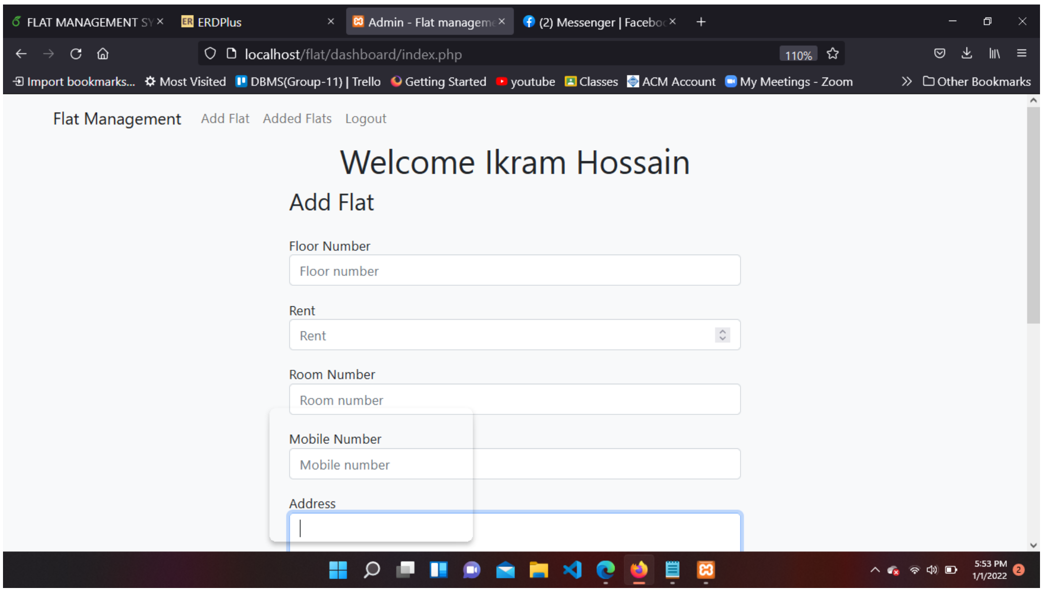
Flat Management

Short description about flat management


FLAT MANAGEMENT SY
ERDPlus
Admin - Flat managem
(2) Messenger | Facebo
localhost/flat/dashboard/index.php
110%
Import bookmarks... Most Visited DBMS(Group-11) | Trello Getting Started youtube Classes ACM Account My Meetings - Zoom Other Bookmarks
Flat Management
Dashboard
Logout
Show 10 entries
Search:
5:52 PM
1/1/2022

Book Flat

ID	Type	Address	Floor No.	Rent	Room	Facilities
	Type x	Address x	Floor x	12000 18100 x	2 10 x	
#11	Flat	Muradpur,chittagong Mobile: 0193832930 Booked	3	12000	6	Wifi Ac
#12	Flat	Hathazari,Chittagong Mobile: 01572012938 Cancel Booking	2	15000	7	Wifi Ac
#13	Flat	hhh Mobile: 222 Request for Booking	7	12000	2	Wifi Ac


FLAT MANAGEMENT SY
ERDPlus
Admin - Flat manageme
(2) Messenger | Facebo
localhost/flat/dashboard/index.php
110%
Import bookmarks... Most Visited DBMS(Group-11) | Trello Getting Started youtube Classes ACM Account My Meetings - Zoom
Other Bookmarks
Flat Management
Add Flat
Added Flats
Logout
5:53 PM
1/1/2022

Welcome Ikram Hossain

Add Flat

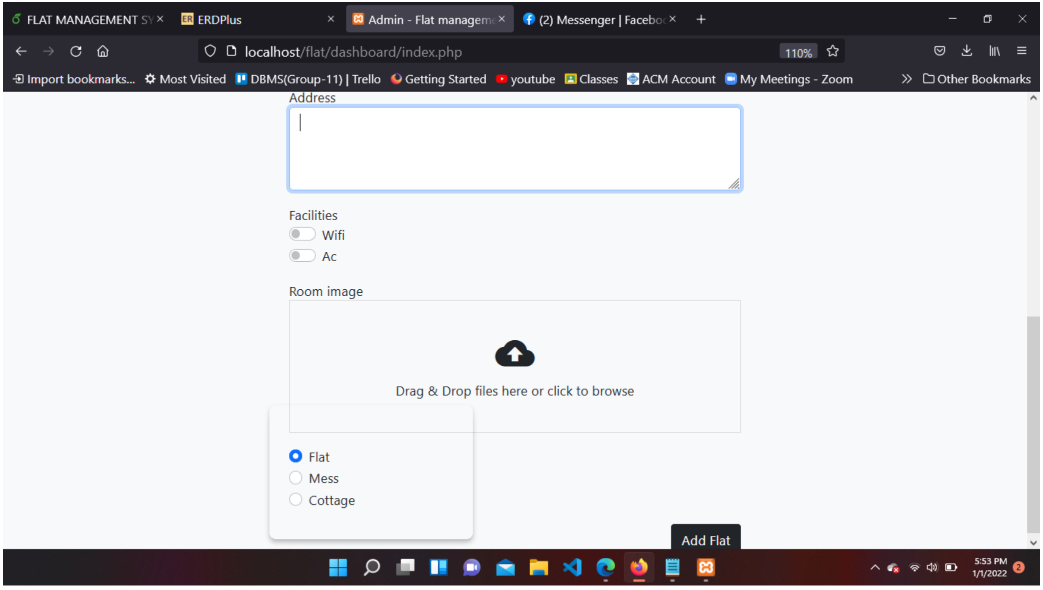
Floor Number: Floor number

Rent: Rent

Room Number: Room number

Mobile Number: Mobile number

Address

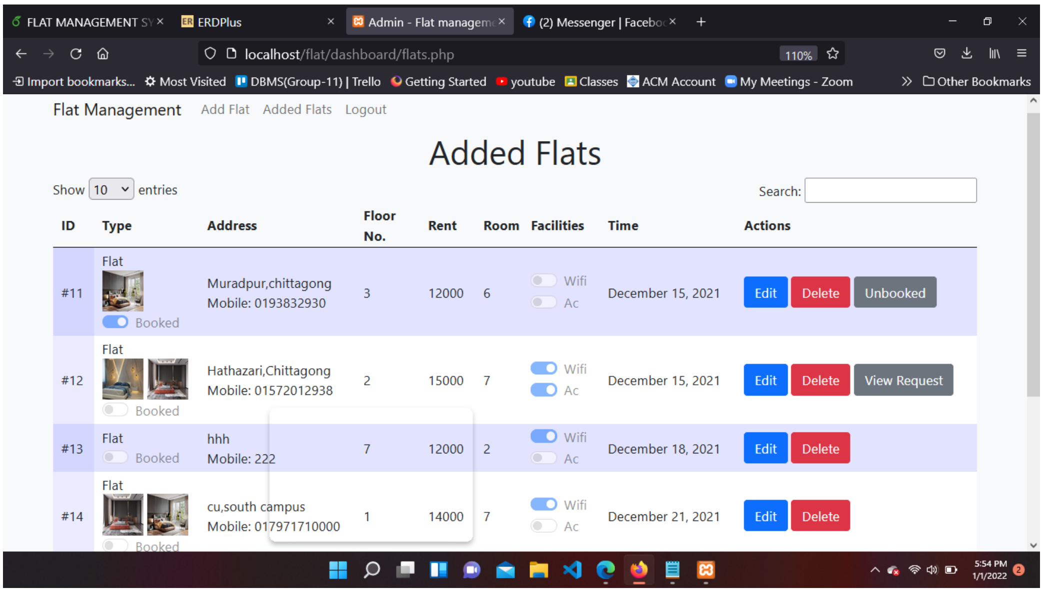

FLAT MANAGEMENT SY
ERDPlus
Admin - Flat manageme
(2) Messenger | Facebo
localhost/flat/dashboard/index.php
110%
Import bookmarks... Most Visited DBMS(Group-11) | Trello Getting Started youtube Classes ACM Account My Meetings - Zoom
Other Bookmarks
Drag & Drop files here or click to browse
Add Flat
5:53 PM
1/1/2022

Address

Facilities
- Wifi
- Ac

Room image

● Flat
○ Mess
○ Cottage



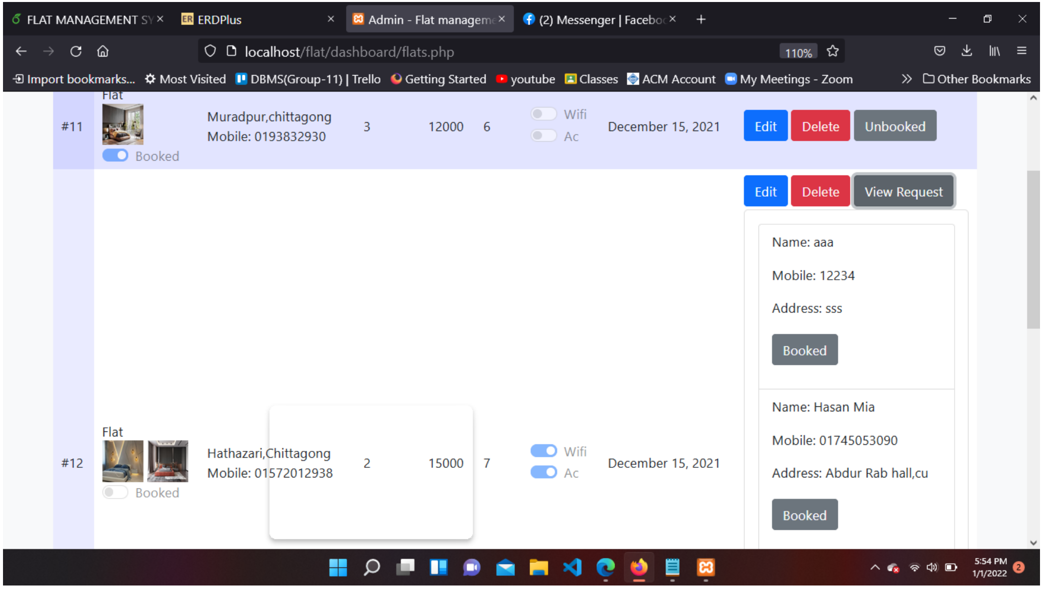
Flat Management   Add Flat   Added Flats   Logout

Added Flats

Show 10 entries

Search:

ID	Type	Address	Floor No.	Rent	Room	Facilities	Time	Actions
#11	Flat Booked	Muradpur,chittagong Mobile: 0193832930	3	12000	6	Wifi Ac	December 15, 2021	Edit Delete Unbooked
#12	Flat Booked	Hathazari,Chittagong Mobile: 01572012938	2	15000	7	Wifi Ac	December 15, 2021	Edit Delete View Request
#13	Flat Booked	hhh Mobile: 222	7	12000	2	Wifi Ac	December 18, 2021	Edit Delete
#14	Flat Booked	cu,south campus Mobile: 017971710000	1	14000	7	Wifi Ac	December 21, 2021	Edit Delete



#11	Flat Booked	Muradpur,chittagong Mobile: 0193832930	3	12000	6	Wifi Ac	December 15, 2021	Edit Delete Unbooked
#12	Flat Booked	Hathazari,Chittagong Mobile: 01572012938	2	15000	7	Wifi Ac	December 15, 2021	Edit Delete View Request

Name: aaa

Mobile: 12234

Address: sss

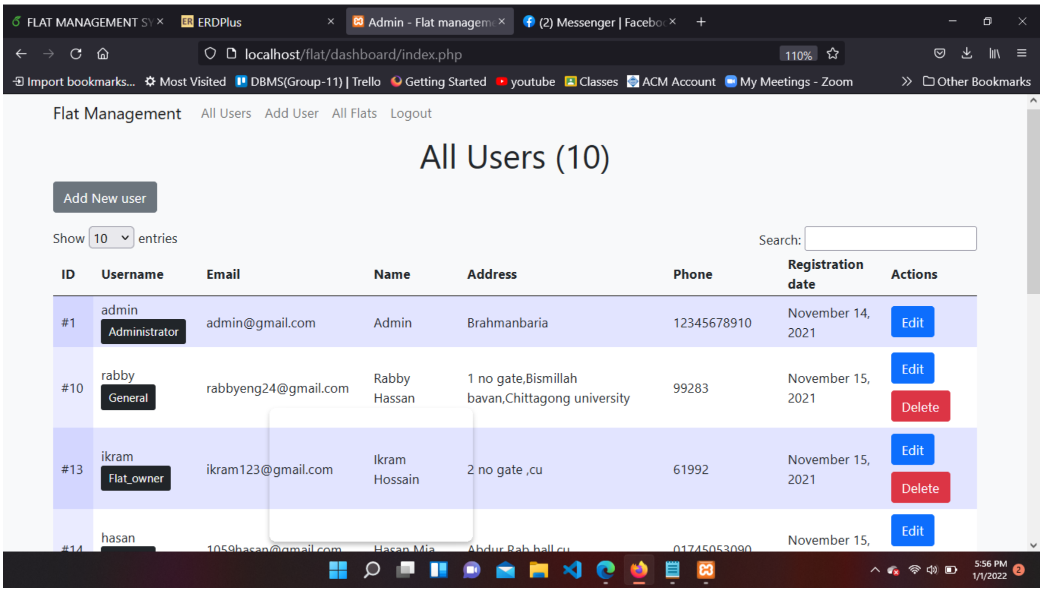
Booked

Name: Hasan Mia

Mobile: 01745053090

Address: Abdur Rab hall,cu

Booked


FLAT MANAGEMENT SY
ERDPlus
Admin - Flat manageme
(2) Messenger | Facebo
localhost/flat/dashboard/index.php
110%
Import bookmarks... Most Visited DBMS(Group-11) | Trello Getting Started youtube Classes ACM Account My Meetings - Zoom Other Bookmarks
5:56 PM
1/1/2022

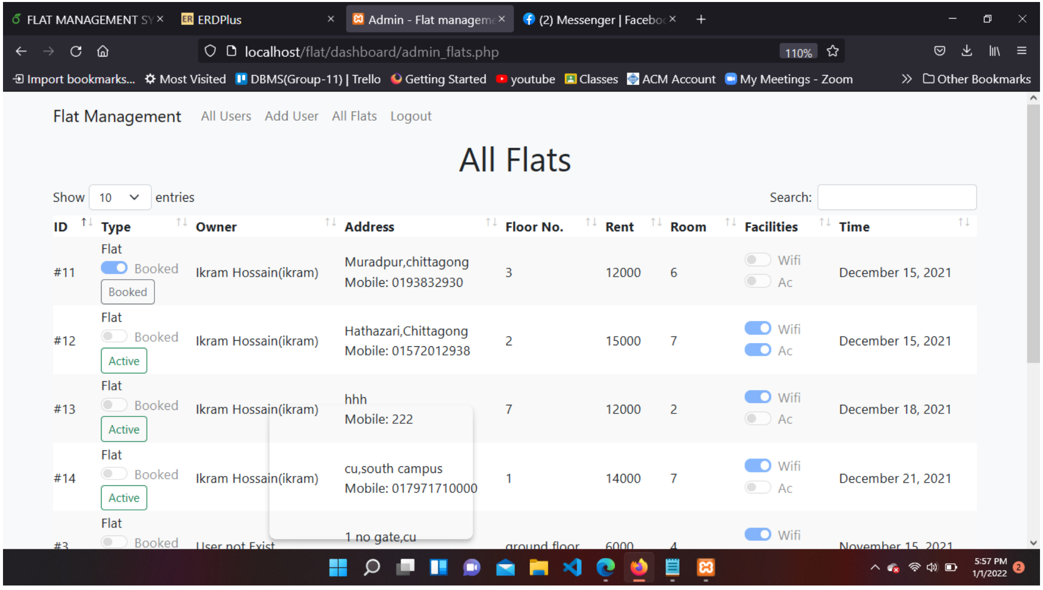
Flat Management All Users Add User All Flats Logout

All Users (10)

Add New user

Show 10 entries Search:

ID	Username	Email	Name	Address	Phone	Registration date	Actions
#1	admin Administrator	admin@gmail.com	Admin	Brahmanbaria	12345678910	November 14, 2021	Edit
#10	rabby General	rabbyeng24@gmail.com	Rabby Hassan	1 no gate,Bismillah bavan,Chittagong university	99283	November 15, 2021	Edit Delete
#13	ikram Flat_owner	ikram123@gmail.com	Ikram Hossain	2 no gate ,cu	61992	November 15, 2021	Edit Delete
#14	hasan	1059hasan@gmail.com	Hasan Mia	Abdur Rab hall,cu	01745053090	November 15,	Edit

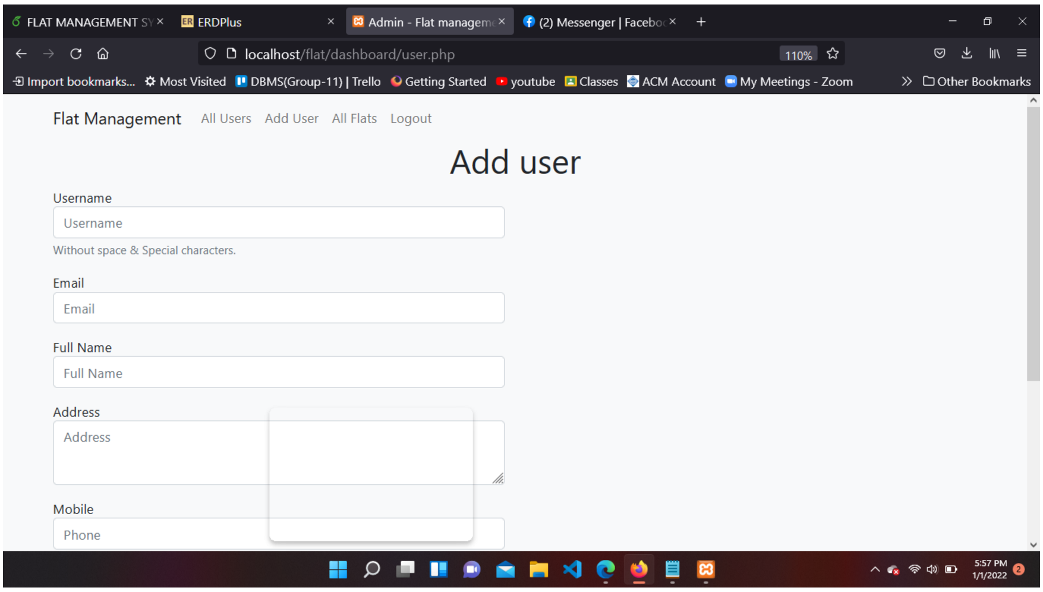

FLAT MANAGEMENT SY
ERDPlus
Admin - Flat manageme
(2) Messenger | Facebo
localhost/flat/dashboard/admin_flats.php
110%
Import bookmarks... Most Visited DBMS(Group-11) | Trello Getting Started youtube Classes ACM Account My Meetings - Zoom Other Bookmarks
5:57 PM
1/1/2022

Flat Management All Users Add User All Flats Logout

All Flats

Show 10 entries Search:

ID	Type	Owner	Address	Floor No.	Rent	Room	Facilities	Time
#11	Flat ● Booked Booked	Ikram Hossain(ikram)	Muradpur,chittagong Mobile: 0193832930	3	12000	6	○ Wifi ○ Ac	December 15, 2021
#12	Flat ○ Booked Active	Ikram Hossain(ikram)	Hathazari,Chittagong Mobile: 01572012938	2	15000	7	● Wifi ● Ac	December 15, 2021
#13	Flat ○ Booked Active	Ikram Hossain(ikram)	hhh Mobile: 222	7	12000	2	● Wifi ○ Ac	December 18, 2021
#14	Flat ○ Booked Active	Ikram Hossain(ikram)	cu,south campus Mobile: 017971710000	1	14000	7	● Wifi ○ Ac	December 21, 2021
#3	Flat ○ Booked	User not Exist	1 no gate,cu	ground floor	6000	4	● Wifi	November 15, 2021



Add user

Username

Username

Without space & Special characters.

Email

Email

Full Name

Full Name

Address

Address

Mobile

Phone

11 Software Deployment

11.1 Server setting up

We have used localhost, Apache can be the best option for this. Firstly we have to configure XAMPP with Apache and Mysql.

11.2 Database setting up

After that we have to copy our database in phpMyadmin using exported sql command.

11.3 Creating shortcut

Then we can run our php files in browser.For general user we can create shortcut for this system which will appear in home screen of the users device.

11.4 Run the application

Finally any non-technical user can run this application by clicking the created shortcut.

12 Conclusion and Future Work

We wish that our Flat Management System will be user friendly. So we want to add a payment system to our Flat Management System in future. Besides this we want to activate messaging service in our project. Furthermore we want to add a review and a rating system so that users can express their opinion.

13 Bibliography

[1] [2] [3]

References

[1] Lynn Beighley. *Head First SQL: Your Brain on SQL–A Learner's Guide.* " O'Reilly Media, Inc.", 2007.

[2] Abraham Silberschatz, Henry F Korth, Shashank Sudarshan, et al. *Database system concepts*, volume 4. McGraw-Hill New York, 1997.

[3] John Watson. *OCA Oracle Database 11g: Administration 1 Exam Guide (exam 1Z0-052).* McGraw-Hill/Osborne, 2008.