**Name: ZESHAN SAYED**

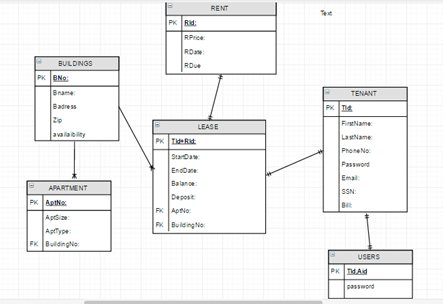
**ID:20379978341**

**Description:**

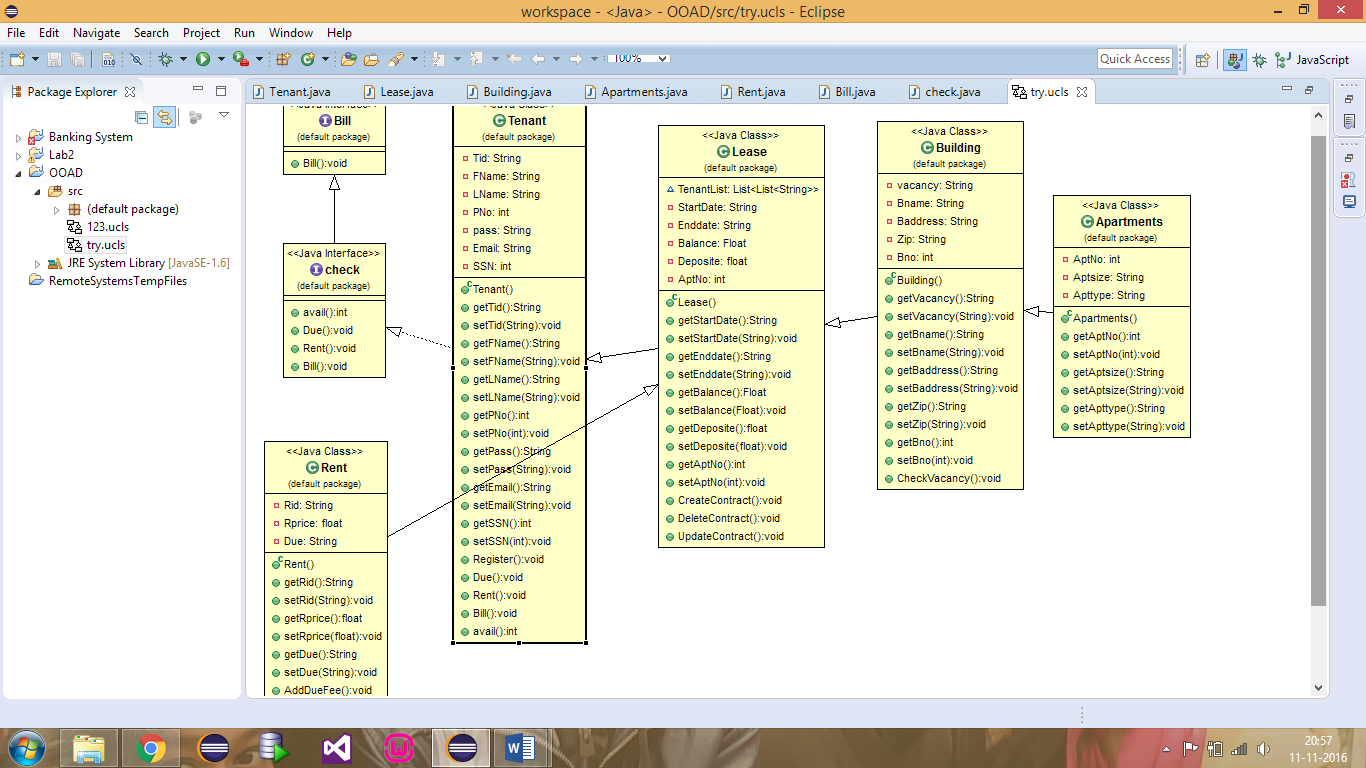
The proposed system is an Apartment Management System which incorporates CRUD operations effectively to have integrity in the system. The users checks in to the system and search for availability if yes then registers. Then admin creates user and assigns rent, lease and deposits. After lease ends admin updates the lease according to termination and revival.

The system consist of tenants as input to the table and bills and availability as the output. At first the user check availability if yes then logs in to the system and registers, then they can view the respective bills and due dates. The table Tenant consist of creating, updating and deleting tenants from the table which indirectly affects the vacancy. The table Lease consist of admin view and tenant view where admin generates the bill while tenant follows the procedure. The Lease class consist of Create, Update and Delete contract. The Building class is the parent class (inheritance) and have different apartments as Apartment class. Finally the rent is generated with interface Bill and printed on to the Tenants Class which inherits lease class (inheritance).So basically it follows CRUD operation after checking availability in the system and follows DDL and DML operations.

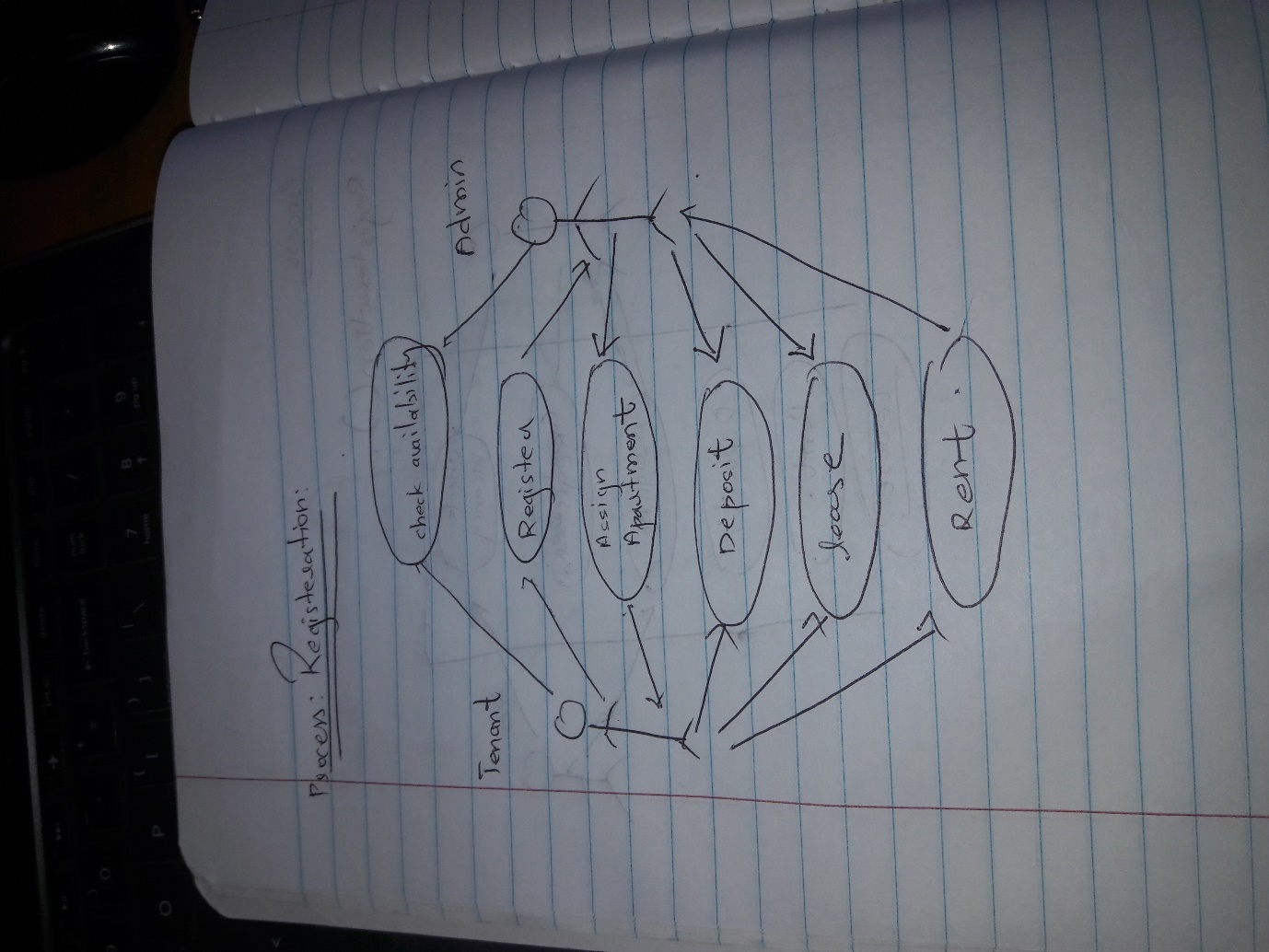
**ERD:**



CLASS DIAGRAM:



**Use Case:**



**DDL STATEMENTS:**

***TENANT TABLE:***

create table TENANT(

tid varchar(20) not null primary key,

Fname varchar(20),

LName varchar(20),

PhoneNo number,

password varchar(20),

email varchar(20),

SSN number,

Bill float);

***USER TABLE:***

create table USER(

tid varchar(20) not null,

aid varchar(20) not null,

password varchar(20) not null,

Primary key (tid,aid),

Foreign key (tid) references TENANT(tid));

***RENT TABLE:***

create table RENT(

rid varchar(20) not null primary key,

Rprice float,

Rdate date,

Rdue date);

***LEASE TABLE:***

create table LEASE(

rid varchar(20) not null,

tid varchN(20) not null

StartDate date,

CEndDate date,

Balance float,

AptNo number

BuildingNo number

Foreign key(AptNo) references Apartment(AptNo),

Foreign key(BuildinNo)references Buildings(BuildingNo),

Foreign key(rid) references RENT(rid),

Foreign key(tid) references TENANT(tid),

Primary key(rid,tid));

***Buildings Table:***

create table BUILDINGS(

BNo varchar(20) not null primary key,

Bname varchar(20),

BAdress varchar(50),

ZIP varchar(10),

availaibility number);

***Apartment table:***

create table APARTMENT(

AptNo number not null primary key,

AptSize varchar(20),

AptType varchar(10),

BuildingNo number,

Foreign key (BuildingNo) references BUILDINGS (BuildingNo));