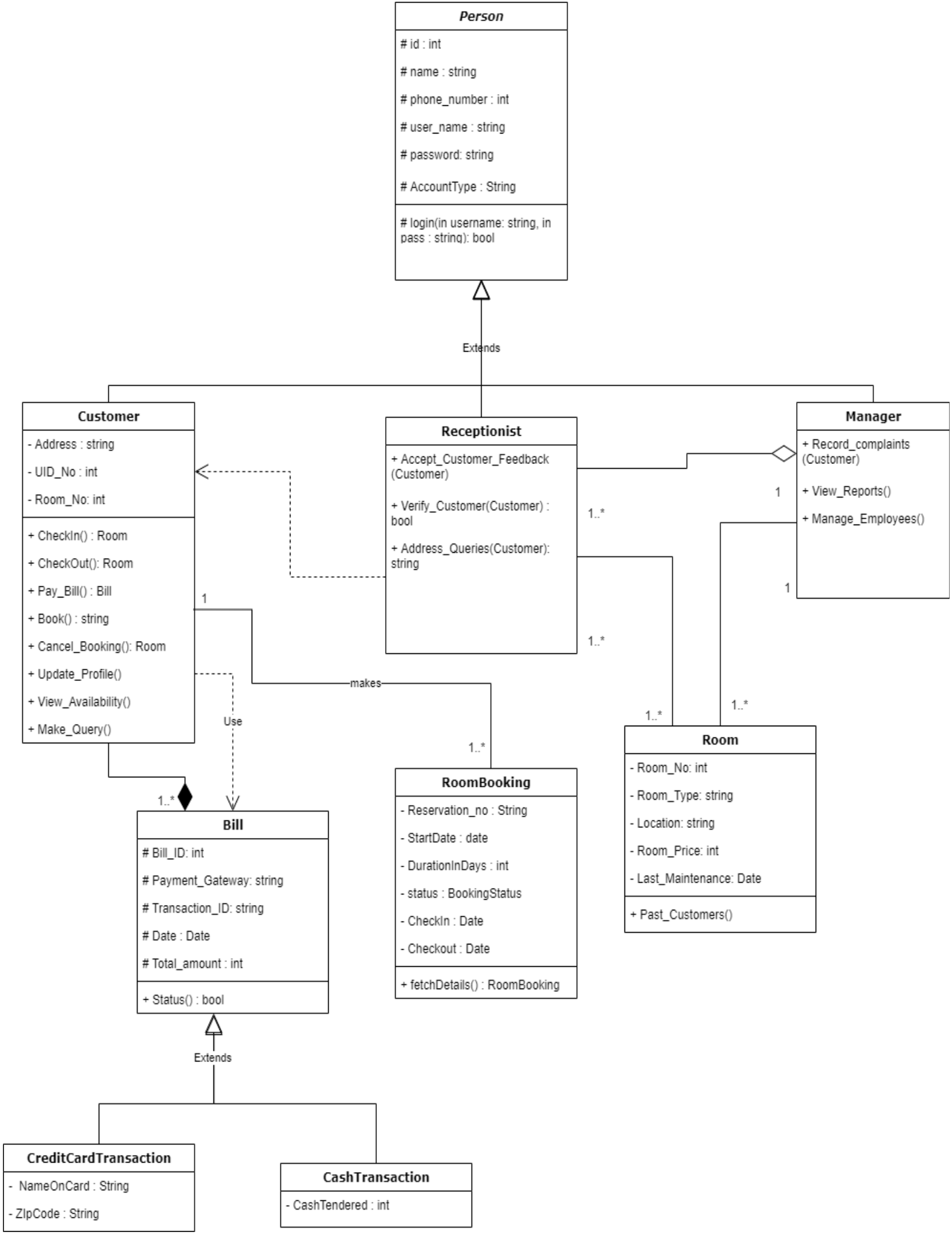
Design Model-Class Model for Hotel Management System

Name	Roll Number	Gr. Number	Div/Batch
Atharva Parikh	332044	21910716	B/B2
Sidhant Khamankar	332029	21910598	B/B2
Namrata Kadav	332040	21911024	B/B2

Objective

Study in detail working of system/Project.
Identify Design classes/ Evolve Analysis Model. Use advanced relationships.
Draw Design class Model using OCL and UML 2.0 Notations.
Implement the design model with a suitable object-oriented language.

Diagram:


Person
id : int
name : string
phone_number : int
user_name : string
password: string
AccountType : String
login(in username: string, in pass : string): bool
Extends
Customer
- Address : string
- UID_No : int
- Room_No: int
+ CheckIn() : Room
+ CheckOut(): Room
+ Pay_Bill() : Bill
+ Book() : string
+ Cancel_Booking(): Room
+ Update_Profile()
+ View_Availability()
+ Make_Query()
Receptionist
+ Accept_Customer_Feedback (Customer)
+ Verify_Customer(Customer) : bool
+ Address_Queries(Customer): string
Manager
+ Record_complaints (Customer)
+ View_Reports()
+ Manage_Employees()
1
1..*
1..*
1
makes
1..*
Use
1..*
1..*
1..*
Bill
Bill_ID: int
Payment_Gateway: string
Transaction_ID: string
Date : Date
Total_amount : int
+ Status() : bool
RoomBooking
- Reservation_no : String
- StartDate : date
- DurationInDays : int
- status : BookingStatus
- CheckIn : Date
- Checkout : Date
+ fetchDetails() : RoomBooking
Room
- Room_No: int
- Room_Type: string
- Location: string
- Room_Price: int
- Last_Maintenance: Date
+ Past_Customers()
Extends
CreditCardTransaction
- NameOnCard : String
- ZIpCode : String
CashTransaction
- CashTendered : int

Code:

```
import java.util.*;

public class Design {
    public class Person {
        protected int id;
        protected int ph_no;
        protected String name;
        protected String username;
        protected String Password;
        protected String account_type;

        protected boolean login(String username, String password) {
            // .....
            if (username == this.username && password == this.Password)
                return true;

            return false;
        }
    };

    public class Customer extends Person {
        // following variables make up the userdata
        String Address;
        int UID_NO;
        int RoomNo;

        // boolean CheckIn(userdata){
        // //logic to save data to session storage or as cookies
        // if (data saved successfully){
        // return true;
        // }
        // return false; //some problem occured
        // }

        // void CheckOut(userdata){
        // //logic to remove user session data from device and browser
        // // call to display home screen;
```

```
// }

Bill payBill() {
    // call to payment gateway for bill payment
    // bill = call to respective function of bill object
    Bill b = new Bill();
    if (b.Status()) {
        return b;
    }
    return null;
}

// int Book(userdata){
// //display all available rooms
// //attempt to book a room in the hotel
// //call to payBill function
// boolean roomBooked = book provided room;
// if(roomBooked){
// return 1;
// }
// else{
// return -1; //server error
// }
// }

// int CancelBooking(userdata){
// //search for user and remove the booking
// //apply the cancellation charges according to rules
// boolean userFound = find user;
// boolean dataRemoved = removeData;
// if(userFound && dataRemoved){
// return amount; //amount to be returned to user
// }
// else{
// return -1; //error
// }
// }

// boolean UpdateProfile(userdata){
```

```
        // //search for a user
        // if(user found){
        // //update data according to what user wants
        // return true;
        // }
        // else{
        // return false; //user doesn't exist
        // }
        // }

        // void ViewAvailability(roomData){
        // //search for rooms according to roomData provided
        // // boolean roomsfound = search for rooms;
        // if(roomsfound){
        // //display the rooms
        // if(interested){
        // //call for booking
        // }
        // }
        // else{
        // //show next available booking period
        // //call for book function
        // }
        // }
    };

    public class Manager extends Person {
        public void Record_complaints(Customer c) {
            // Store complaints from a customer
            Receptionist rep = new Receptionist();
            System.out.println(rep.Address_Queries(c));
        }

        public void View_Reports() {
            // View financial reports using Billing information
            Rooms R = new Rooms();
            for (Customer c : R.past_customers()) {
                System.out.println(c.payBill());
            }
```

```
    }

    public void Manage_Employees() {
        // Manage Employee leaves, payment, work assignment
    }
}

public class Receptionist extends Person {
    public void Accept_Customer_Feedback(Customer c) {
        // Accept the feedback and store it in database
        // along with customer id

    };

    public Boolean Verify_Customer(Customer c) {
        Customer cust = c;
        class Local {
            public Boolean verifiedCustomer(Customer cust) {
                // Check for the Documents and booking details from customer
                // Check CheckIn date and time
                // Check for advance payment

                Boolean found = false;
                if (found == true) {
                    return true;
                }
                return false;
            }
        }
        if (new Local().verifiedCustomer(cust) == true) {
            return true;
        }
        return false;
    };

    public String Address_Queries(Customer c) {
        String query_status = "Not_resolved!";
        Boolean resolved_Query = false;
        // try to resolve the query ..acordingly update the customer.
```

```
            if (resolved_Query) {
                query_status = "Your query has been resolved !";
            } else {
                query_status = "Sorry for inconveience, we will resolved your query shortly
!";
            }
            // return the status of query resolved or not or in process.
            return query_status;
        };

    }

    enum Status {
        Booked, Pending, Cancelled, CheckIn, CheckOut
    }

    public class RoomBooking {
        private String reservation_no;
        private Date StartDate;
        private int Duration_in_days;

        private Status st;

        private Date CheckIN;
        private Date CheckOut;

        public RoomBooking fetchDetails(String reservation_no) {
            RoomBooking r1 = new RoomBooking();

            return r1;
        }
    }

    public class Rooms {

        private int room_no;
        private String room_type;
        private String location;
        private int Room_price;
```

```
        private Date Last_Maintenance;

        public ArrayList<Customer> past_customers() {
            // Past customers of this Room
            ArrayList<Customer> arr = new ArrayList<Customer>(10);
            return arr;
        }
    }

    public class Bill {
        protected int BillID;
        protected boolean PaymentStatus;
        protected String TransactionID;
        protected Date date;
        protected int TotalAmount;

        boolean Status() {
            // implement the payment gateway
            // PaymentStatus = call to gateway;
            if (PaymentStatus == true) {
                // create the bill
                // show bill and allow printing and saving
                return true;
            } else {
                // ask user to try again
                return false;
            }
        }
    };

    public class CashTransaction extends Bill {
        String Name_on_card;
        String Zipcode;
    }

    public class CreditCardTransaction extends Bill {
        int cash_tendered;
    }
};
```