Page: प्रगति®
Date:

VehicleRentalSystem {

List<Store> storelist;
List<User> userlist;

addUser();
RemoveUser();
addStore()
removeStore();
getStore()

}.

Data Flow -

Add user & stores.
User search the location.
get All store of that location.
get All vehicle of ~~that~~ particular store
Create Reservation at store;
Generate Bill
Make payment
Complete Reservation at store.

11)

```
Store {
    int storeId;
    VehicleInventoryManagemet ;
    Location storelocation;
    List <Reservation> ;

    List<Vehicle> getVehicle (VehicleType);
    void setVehicle (List<Vehicle> );

    Reservation CreateReservation ( );
    boolean CompleteReservation ( int ReservationId );
    boolean updateReservation (int ReservId );
}.
```

12)

```
User {.
    int userId;
    String name;
    String license Number;
}.
```

13)

```
VehicleInventoryManagement {
    List <Vehicle > vehicles
}.
```

Page: प्रगति
Date:

~~LLD~~ with ~~vending~~ Machine.

LLD of Car Booking App

1) Vehicle {

int vehicleId;
vehicle no.;
VehicleType;
Company Name;
Model Name;
KM Osiven;
Average;
CC;
daily RentalCost;
Hourly RentalCost;
no of ~~ste~~ seat;
Status;

// getter setter

}.

2) enum VehicleType {. Car, bike, Scoty }.
3) enum Status {. Active, InActive }.

8) Reservation {
int Id;
User user;
Vehicle vehicle;
Date BookingDate;
Date BookingDateFrom;
Date BookingDateTo;
Time fromTimeStamp;
Time toTimeStamp;
Location pickupLocation;
Location dropLocation;
ReservationType;
ReservationStatus;

// CRUD operation
create();
Read()
Update()
Delete();

}.

ReservationType { Hourly, Daily }
ReservationStatus { Scheduled, ~~Comp~~ Booked, Cancelled }.
{ Pending, Confirm, Cancelled }.

4) Bill {

Reservation ;
total Bill amount ;
boolean isbillpaid ;
double ComputeBill() { } ;

}

5) Payment {
int PaymentId ;
int amountpaid ;
Date dateofpayment ;
~~boolean isRefundable ;~~
Payment mode
Payment status ;

void payBill (Bill bill) ;

}

6) Enum payment mode { card, cash }

7) Enum payment status { unpaid, pending, completed
Refund } ;