CAR PARKING SYSTEM

Name : Dinesh Ganesan
CWID : A20369997
Course : ITMD 510 – Object Oriented App Devp
Project : Phase-I
Date : 04/07/2016

Table of Contents

1. DESCRIPTION OF THE PROJECT

Car Parking System is an application where a user can reserve a car parking in any location (i.e., location which are enrolled in the system). Based on how long the car has been parked in the parking lot, the amount will be calculated which will be taken care in the future system. There are two types of user who uses this system

- System Admin
- User

System Admin – There are many operations which the system admin and admins is the one who takes care of adding, removing, altering (basically CRUD operations) related to the Parking System.

User – The major operation which user does is reserves a parking lot in any location and pays of the Parking.

2. FUNCTIONALITIES OF THE USER

2.1. System Admin

- Login
- Authorizes Parking Lot
- Authorizes Users
- Add parking locations
- Adds parking spaces in the parking location
- Updates the parking space in the parking location
- Deletes and updates if the parking lot is under construction or if any repair
- Adds the layout of the parking space.

2.2. User

- Login
- Reverser a parking space
- Pays for the Parking space used.

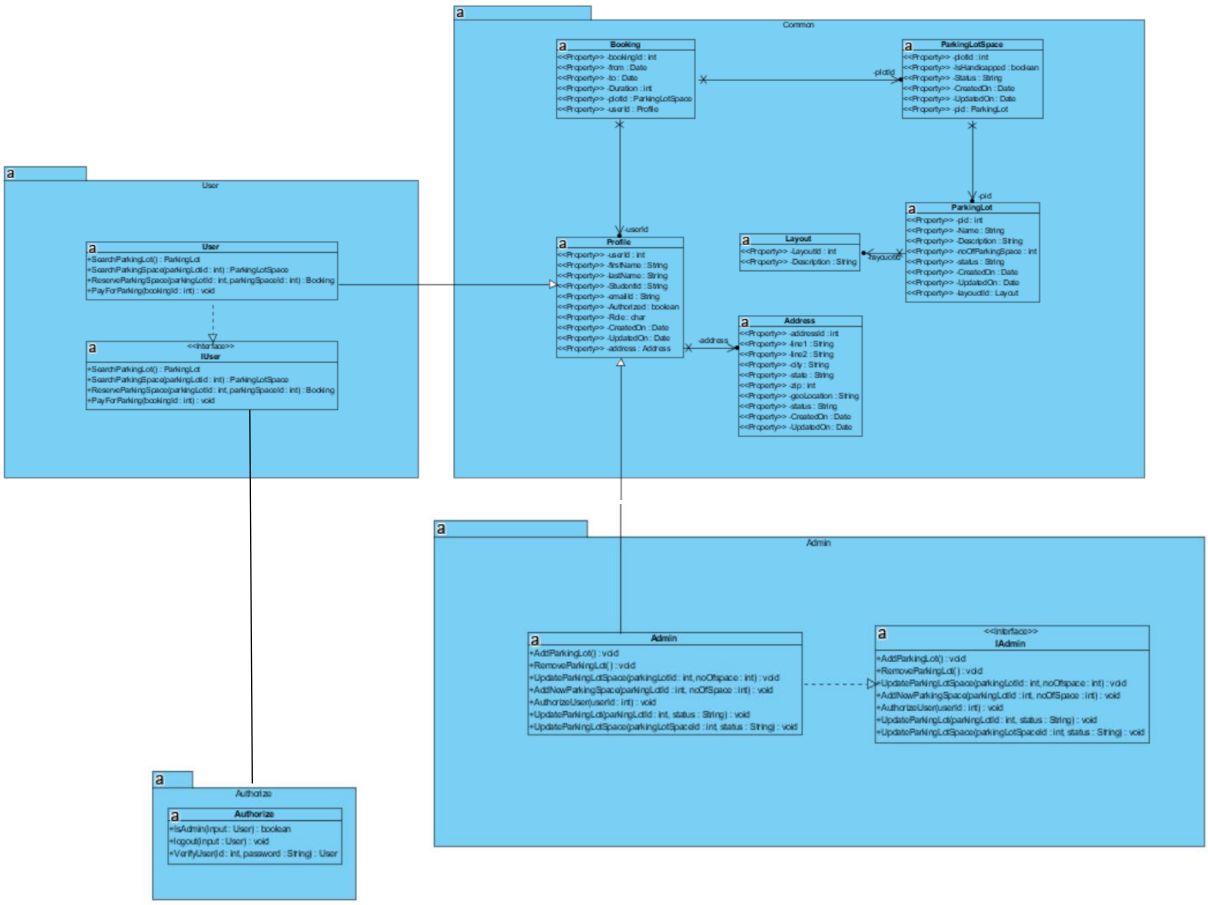
3. CLASS DESCRIPTION

- Profile is the base class for the User and Admin Class, and profile class holds the basic information of the profile details.
- User Class implements the IUser Interface, which deals with the user related operations like
  - ✓ Searching Parking Lot
  - ✓ Searching Parking Space
  - ✓ Reserve Parking Space
  - ✓ Pay for Parking space Used (will be covered in future)
- Admin class implements the IAdmin Interface, which deals with the following functionalities
  - ✓ Add Parking Lot
  - ✓ Remove Parking Lot

- ✓ Update Parking Space
- ✓ Add additional parking space
- ✓ Authorize User
- ✓ Update Parking Lot

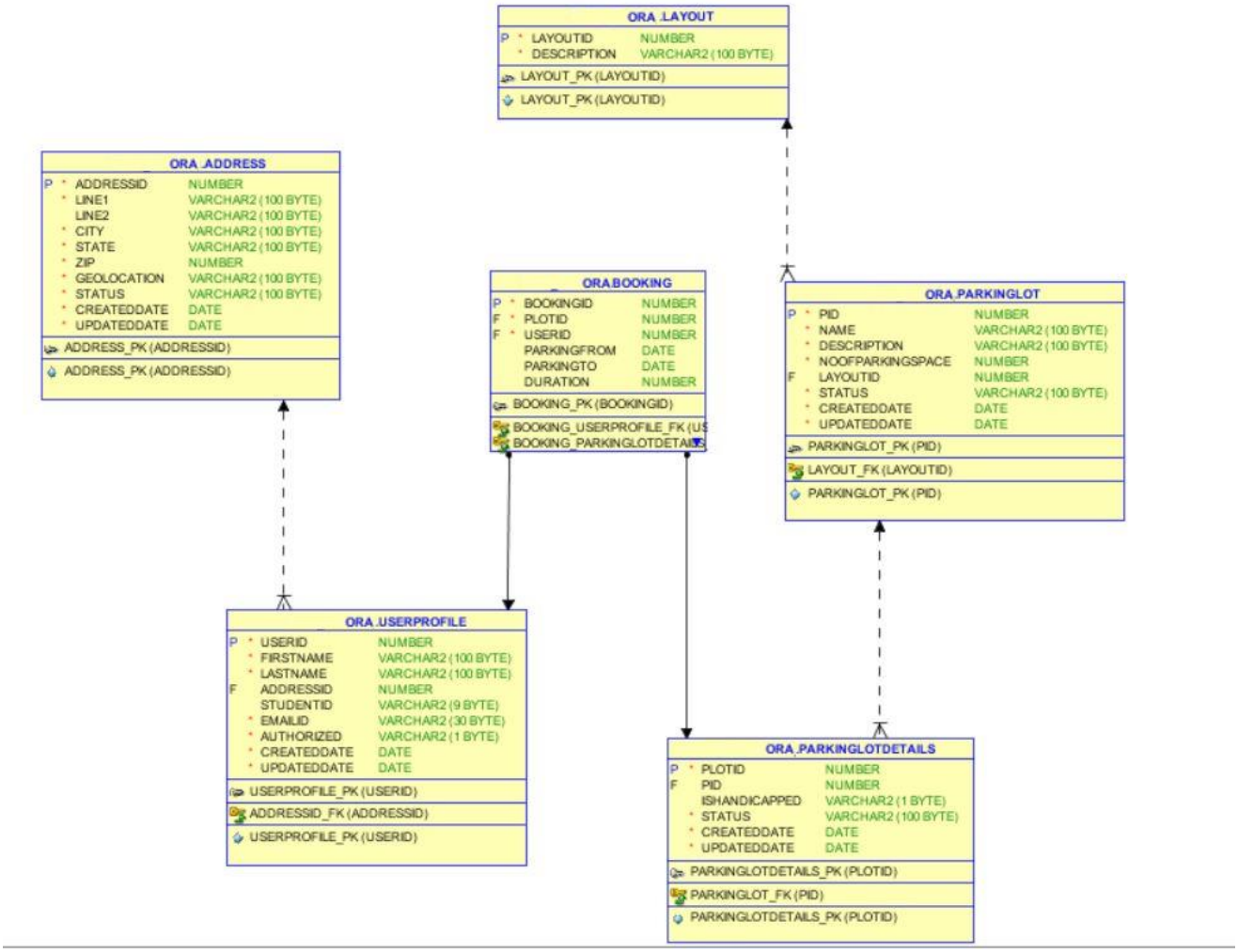
- Authorize class is used for login, checks whether the user is admin or not and for logout purpose. This class is purely independent where only association is dealt.
- Layout class is used to hold the image layout of the parking space, similarly Address, Parking Lot and Parking Lot Space
- Booking class is used to hold the information relating to the reserving or booking the parking space.

3.1. UML Diagram



Note: All the class has its own getter and setter which is not shown in the Class diagram above. Considering its implicitly given.

3.2. ER Diagram



4. DDL STATEMENTS

The following are the DDL Create queries which are used to create the table in the database.

4.1. Address Table

```
CREATE TABLE ADDRESS (ADDRESSID NUMBER,
        LINE1 VARCHAR2(100) NOT NULL,
        LINE2 VARCHAR2(100) DEFAULT NULL,
        CITY VARCHAR2(100) NOT NULL,
        STATE VARCHAR2(100) NOT NULL,
        ZIP NUMBER NOT NULL,
        GEOLOCATION VARCHAR2(100) NOT NULL,
```

```
        STATUS VARCHAR2(100) NOT NULL,
        CREATEDDATE DATE NOT NULL,
        UPDATEDDATE DATE NOT NULL,
        CONSTRAINT ADDRESS_PK PRIMARY KEY(ADDRESSID)
)
```

4.2. Layout Table

```
CREATE TABLE LAYOUT(
        LAYOUTID NUMBER,
        DESCRIPTION VARCHAR2(100) NOT NULL,
        CONSTRAINT LAYOUT_PK PRIMARY KEY(LAYOUTID)
)
```

4.3. UserProfile Table

```
CREATE TABLE USERPROFILE(
        USERID NUMBER,
        FIRSTNAME VARCHAR2(100) NOT NULL,
        LASTNAME VARCHAR2(100) NOT NULL,
        ADDRESSID NUMBER,
        STUDENTID VARCHAR2(9) DEFAULT NULL,
        EMAILID VARCHAR2(30) NOT NULL,
        AUTHORIZED VARCHAR2(1) NOT NULL,
        CREATEDDATE DATE NOT NULL,
        UPDATEDDATE DATE NOT NULL,
        CONSTRAINT USERPROFILE_PK PRIMARY KEY(USERID),
        CONSTRAINT ADDRESSID_FK FOREIGN KEY (ADDRESSID)
        REFERENCES ADDRESS(ADDRESSID)
)
```

4.4. ParkingLot Table

```
CREATE TABLE PARKINGLOT(
        PID NUMBER,
        NAME  VARCHAR2(100) NOT NULL,
        DESCRIPTION  VARCHAR2(100) NOT NULL,
        NOOFPARKINGSPACE NUMBER NOT NULL,
        LAYOUTID NUMBER,
        STATUS VARCHAR2(100) NOT NULL,
        CREATEDDATE DATE NOT NULL,
        UPDATEDDATE DATE NOT NULL,
        CONSTRAINT PARKINGLOT_PK PRIMARY KEY (PID),
        CONSTRAINT LAYOUT_FK FOREIGN KEY (LAYOUTID)
        REFERENCES LAYOUT(LAYOUTID)
)
```

4.5. ParkingLotDetails Table

```
CREATE TABLE PARKINGLOTDETAILS(
        PLOTID NUMBER,
        PID NUMBER,
        ISHANDICAPPED VARCHAR2(1),
        STATUS VARCHAR2(100) NOT NULL,
        CREATEDDATE DATE NOT NULL,
        UPDATEDDATE DATE NOT NULL,
        CONSTRAINT PARKINGLOTDETAILS_PK PRIMARY KEY (PLOTID),
        CONSTRAINT PARKINGLOT_FK FOREIGN KEY (PID) REFERENCES PARKINGLOT(PID)
)
```

4.6. Booking Table

```
CREATE TABLE BOOKING(
        BOOKINGID NUMBER,
        PLOTID NUMBER,
        USERID NUMBER,
        PARKINGFROM DATE,
        PARKINGTO DATE,
        DURATION NUMBER,
        CONSTRAINT BOOKING_PK PRIMARY KEY (BOOKINGID),
        CONSTRAINT PARKINGLOTDETAILS_FK FOREIGN KEY (PLOTID) REFERENCES
        PARKINGLOTDETAILS(PLOTID),
        CONSTRAINT USERPROFILE_FK FOREIGN KEY (USERID) REFERENCES USERPROFILE(USERID)
)
```

5. Entities Files

Few entities are given below and the below entity code is just the skeleton of the entity, not the entire code.

5.1. Address Entity

```
package Common;

import java.util.*;

public class Address {
	private int addressId;
	public int getAddressId() {
		return addressId;
	}
	public void setAddressId(int addressId) {
		this.addressId = addressId;
	}
	public String getLine1() {
		return line1;
	}
	public void setLine1(String line1) {
		this.line1 = line1;
	}
```

```
public String getLine2() {
	return line2;
}
public void setLine2(String line2) {
	this.line2 = line2;
}
public String getCity() {
	return city;
}
public void setCity(String city) {
	this.city = city;
}
public String getState() {
	return state;
}
public void setState(String state) {
	this.state = state;
}
public int getZip() {
	return zip;
}
public void setZip(int zip) {
	this.zip = zip;
}
public String getGeoLocation() {
	return geoLocation;
}
public void setGeoLocation(String geoLocation) {
	this.geoLocation = geoLocation;
}
public String getStatus() {
	return status;
}
public void setStatus(String status) {
	this.status = status;
}
public Date getCreatedOn() {
	return CreatedOn;
}
public void setCreatedOn(Date createdOn) {
	CreatedOn = createdOn;
}
public Date getUpdatedOn() {
	return UpdatedOn;
}
public void setUpdatedOn(Date updatedOn) {
	UpdatedOn = updatedOn;
}
private String line1;
private String line2;
private String city;
private String state;
private int zip;
private String geoLocation;
private String status;
```

```
	private Date CreatedOn;
	private Date UpdatedOn;

}
```

5.2. Booking Entity

```
package Common;

import java.util.*;

public class Booking {
	private int bookingId;
	private ParkingLotSpace plotId;
	private Profile userId;
	private Date from;
	private Date to;
	private int Duration;
	public int getBookingId() {
		return bookingId;
	}
	public void setBookingId(int bookingId) {
		this.bookingId = bookingId;
	}
	public ParkingLotSpace getPlotId() {
		return plotId;
	}
	public void setPlotId(ParkingLotSpace plotId) {
		this.plotId = plotId;
	}
	public Profile getUserId() {
		return userId;
	}
	public void setUserId(Profile userId) {
		this.userId = userId;
	}
	public Date getFrom() {
		return from;
	}
	public void setFrom(Date from) {
		this.from = from;
	}
	public Date getTo() {
		return to;
	}
	public void setTo(Date to) {
		this.to = to;
	}
	public int getDuration() {
		return Duration;
	}
	public void setDuration(int duration) {
		Duration = duration;
	}
```

```
}
```

5.3. Layout Entity

```
package Common;

public class Layout {
	private int LayoutId;
	public int getLayoutId() {
		return LayoutId;
	}
	public void setLayoutId(int layoutId) {
		LayoutId = layoutId;
	}
	public String getDescription() {
		return Description;
	}
	public void setDescription(String description) {
		Description = description;
	}
	private String Description;
}
```

5.4. ParkingLayout Entity

```
package Common;

import java.util.*;

public class ParkingLot {
	private int pid;
	private String Name;
	private String Description;
	private int noOfParkingSpace;
	private Layout layouotId;
	private String status;
	private Date CreatedOn;
	private Date UpdatedOn;
	public int getPid() {
		return pid;
	}
	public void setPid(int pid) {
		this.pid = pid;
	}
	public String getName() {
		return Name;
	}
	public void setName(String name) {
		Name = name;
	}
	public String getDescription() {
		return Description;
	}
	public void setDescription(String description) {
```

```
		Description = description;
	}
	public int getNoOfParkingSpace() {
		return noOfParkingSpace;
	}
	public void setNoOfParkingSpace(int noOfParkingSpace) {
		this.noOfParkingSpace = noOfParkingSpace;
	}
	public Layout getLayouotId() {
		return layouotId;
	}
	public void setLayouotId(Layout layouotId) {
		this.layouotId = layouotId;
	}
	public String getStatus() {
		return status;
	}
	public void setStatus(String status) {
		this.status = status;
	}
	public Date getCreatedOn() {
		return CreatedOn;
	}
	public void setCreatedOn(Date createdOn) {
		CreatedOn = createdOn;
	}
	public Date getUpdatedOn() {
		return UpdatedOn;
	}
	public void setUpdatedOn(Date updatedOn) {
		UpdatedOn = updatedOn;
	}

}
```

5.5. ParkingLayoutSpace Entity

```
package Common;

import java.util.*;

public class ParkingLotSpace {
	private int plotId;
	private ParkingLot pid;
	private boolean IsHandicapped;
```

```
	private String Status;
	private Date CreatedOn;
	private Date UpdatedOn;
	public int getPlotId() {
		return plotId;
	}
	public void setPlotId(int plotId) {
		this.plotId = plotId;
	}
	public ParkingLot getPid() {
		return pid;
	}
	public void setPid(ParkingLot pid) {
		this.pid = pid;
	}
	public boolean isIsHandicapped() {
		return IsHandicapped;
	}
	public void setIsHandicapped(boolean isHandicapped) {
		IsHandicapped = isHandicapped;
	}
	public String getStatus() {
		return Status;
	}
	public void setStatus(String status) {
		Status = status;
	}
	public Date getCreatedOn() {
		return CreatedOn;
	}
	public void setCreatedOn(Date createdOn) {
		CreatedOn = createdOn;
	}
	public Date getUpdatedOn() {
		return UpdatedOn;
	}
	public void setUpdatedOn(Date updatedOn) {
		UpdatedOn = updatedOn;
	}

}
```

5.6. Profile Entity

```
package Common;

import java.util.*;

public class Profile {
	private int userId;
```

```
private String firstName;
private String lastName;
private Address address;
private String StudentId;
private String emailId;
private boolean Authorized;
private char Role;
private Date CreatedOn;
private Date UpdatedOn;
public int getUserId() {
	return userId;
}
public void setUserId(int userId) {
	this.userId = userId;
}
public String getFirstName() {
	return firstName;
}
public void setFirstName(String firstName) {
	this.firstName = firstName;
}
public String getLastName() {
	return lastName;
}
public void setLastName(String lastName) {
	this.lastName = lastName;
}
public Address getAddress() {
	return address;
}
public void setAddress(Address address) {
	this.address = address;
}
public String getStudentId() {
	return StudentId;
}
public void setStudentId(String studentId) {
	StudentId = studentId;
}
public String getEmailId() {
	return emailId;
}
public void setEmailId(String emailId) {
	this.emailId = emailId;
}
public boolean isAuthorized() {
	return Authorized;
}
public void setAuthorized(boolean authorized) {
	Authorized = authorized;
}
public char getRole() {
	return Role;
}
public void setRole(char role) {
```

```
			Role = role;
		}
		public Date getCreatedOn() {
			return CreatedOn;
		}
		public void setCreatedOn(Date createdOn) {
			CreatedOn = createdOn;
		}
		public Date getUpdatedOn() {
			return UpdatedOn;
		}
		public void setUpdatedOn(Date updatedOn) {
			UpdatedOn = updatedOn;
		}

}
```