



**Ahsanullah University of Science and Technology (AUST)**  
Department of Computer Science and Engineering

**Project Proposal: Vacation Home Rental System**

Course No.: CSE4126

Course Title: Distributed Database Systems Lab

**Semester: Spring 2022**

**Date of Submission - 22.08.23**

**Submitted To-**

Ms. Zarin Tasnim Shejuti

Ms. Sanzana Karim Lora

**Submitted By-**

**Member 1:**

190204032 : Md. Azmain Mahtab

**Member 2:**

190204044 : Farhana Hossain Swarnali

Lab Group: A2

Year: 4<sup>th</sup>

Semester: 1<sup>st</sup>

Department: CSE

# Vacation Home Rental System

## Project Planning:

The vacation home rental system is a desktop based project where customers can book their accommodation for the trip. The whole management can be controlled using our software.

## Database Schema:

### Global schema:

Customer (user\_id, username, email, registration\_date, age)  
PropertyOwner (owner\_id, username, email, registration\_date, nid\_no, property\_registration\_no)  
Property (property\_id, owner\_id, title, description, address, city, zip\_code, price\_per\_night, max\_guests, amenities, availability\_start\_date, availability\_end\_date, rating)  
Booking (booking\_id, user\_id, property\_id, check\_in\_date, check\_out\_date, total\_price, payment\_status, booking\_status)  
Review (review\_id, user\_id, property\_id, rating, comment, review\_date)

### Fragmentation schema:

Property1 = PJ<sub>(property\_id, owner\_id, title, description, address, city, zip\_code, price\_per\_night, max\_guests, amenities, rating, availability\_start\_date, availability\_end\_date)</sub> Property  
Property2 = SL<sub>city="ctg"</sub> PJ<sub>(property\_id, owner\_id, description, address, city, zip\_code, availability\_start\_date, availability\_end\_date)</sub> Property  
Property3 = SL<sub>city="syl"</sub> PJ<sub>(property\_id, owner\_id, description, address, city, zip\_code, availability\_start\_date, availability\_end\_date)</sub> Property  
Booking1 = Booking SJ<sub>property\_id=property\_id</sub> Property2  
Booking2 = Booking SJ<sub>property\_id=property\_id</sub> Property3  
PropertyOwner1 = PropertyOwner SJ<sub>owner\_id=owner\_id</sub> Property2  
PropertyOwner2 = PropertyOwner SJ<sub>owner\_id=owner\_id</sub> Property3  
Review 1 = Review SJ<sub>property\_id=property\_id</sub> Property2  
Review 2 = Review SJ<sub>property\_id=property\_id</sub> Property3

### Allocation schema :

Property1<sup>1,2</sup>, Property2<sup>1</sup>, Property3<sup>2</sup>, Booking1<sup>1</sup>, Booking2<sup>2</sup>, PropertyOwner1<sup>1</sup>, PropertyOwner2<sup>2</sup>, Review1<sup>1</sup>, Review2<sup>2</sup>

## Functionalities:

### 1. Customer's register and login:

Here a customers can register by giving their info which is later used to login credentials.

```
SQL> @"C:\Users\Azmain\OneDrive\Documents\VS Code\DDS Lab\Airbnb\PROJECT_login_package.sql"
email: az@gmail.com
pass: Aa1234Bb

Trigger created.

Package created.

Package body created.

Login done...welcome Rahim
```

```
SQL> @"C:\Users\Azmain\OneDrive\Documents\VS Code\DDS Lab\Airbnb\PROJECT_register_package.sql"
reg_name: Rahim
reg_email: az@gmail.com
reg_pass: Aa1234Bb
reg_age: 23
reg_address: Dhaka

Trigger created.

Package created.

Package body created.

reg completed

PL/SQL procedure successfully completed.
```

### 2. Admin login:

Admins can login giving admin mail and password.

```
SQL> @"E:\4.1\DDS Lab\Pro\final\admin_login.sql"
email: admin@gmail.com
pass: 123

Package created.

No errors.

Package body created.

Logged In

PL/SQL procedure successfully completed.
```

### 3. Customer password update:

Customers can update their password by giving info about previous password.

```
SQL> @"E:\4.1\DDS Lab\Pro\final\update_password.sql"
email: a@gmail.com
pass: 123

Trigger created.

Package created.

No errors.

Package body created.

No errors.
Enter value for newpass: 23
email ok
Pass changed

PL/SQL procedure successfully completed.

No errors.
```

CUS_ID	CUS_NAME	CUS_EMAIL	CUS_PASS	CUS_AGE
-----				
CUS_ADDRESS				
-----				
1	A	a@gmail.com	23	20
Dhaka				
2	B	b@gmail.com	123	25
Dhaka				

### 4. Search for property:

Customers can search for property.

```
checkin_time: 2023-08-25
checkout_time: 2023-08-30

Package created.

Package body created.

=====
=> Property ID: 1
=> Property Name: abc
=> Price Per Night: 1000
=> Max No. of Guests: 5
=====
=> Property ID: 2
=> Property Name: abc
=> Price Per Night: 1200
=> Max No. of Guests: 2
=====
```

## 5. Book:

Customers can book accommodation from the software.

```
SQL> @"C:\Users\Azmain\OneDrive\Documents\VS Code\DDS Lab\Airbnb\PROJECT_book_property_package.sql"
Property_id: 1
User Email: a@gmail.com
Checkin_time: 2023-08-25
Checkout_time: 2023-08-30
Total_guests: 4

Package created.

Package body created.

=====
Amount To Pay: 4500
Nights Booked: 5
Discount: 100
Insert Booking ID: 7 to Complete Payment
=====
```

## 6. Payment:

Payment for the accommodation is verified here.

```
SQL> @"E:\4.1\DDS Lab\Pro\final\payment.sql"

Package created.

Package body created.

Enter value for book_id: 1
Enter value for pin: 4444
Enter value for property_id: 2
Payment Completed
```

## 7. Personal booking check:

Customers can see their booking history.

```
SQL> @"E:\4.1\DDS Lab\Pro\final\personal_book_check.sql"

Package created.

Package body created.

No errors.
Enter value for customer_id: 1
book_id po_id property_id check_in check_out pay_status book_status
1 2 2 16-AUG-23 18-JUN-23 done Act
```

## 8. Book Cancel:

Here customers can cancel their booking.

```
SQL> @"E:\4.1\DDS Lab\Pro\final\book_cancel.sql"

Package created.

Package body created.

Enter value for book_id: 1
Enter value for customer_id: 1
Booking Canceled

PL/SQL procedure successfully completed.

No errors.
SQL> select * from Booking;
```

BK_ID	PO_ID	PRP_ID	CUS_ID	CHECK_IN	CHECK_OUT	PAY_S	BOOK_
1	2	2	1	16-AUG-23	18-JUN-23	done	Nact

## 9. Admin Property Add:

Admins can add property.

```
SQL> @"E:\4.1\DDS Lab\Pro\final\admin_propertyAdd.sql"
email: admin@gmail.com
pass: 123

Package created.

No errors.

Package body created.

No errors.
Enter value for po_id: 2
Enter value for prptitle: tile
Enter value for prpdesc: desc
Enter value for prp_address: Syl
Enter value for city: Syl
Enter value for zip_code: 123
Enter value for price_night: 1200
Enter value for max_guests: 2
ok
Property Added

PL/SQL procedure successfully completed.

No errors.
SQL> Select * from property;
```

```
SQL> Select * from property;
```

PRP_ID	PO_ID	PRP_T	PRP_D	PRP_ADDRESS	CITY	ZIP_C
1	2	abc	xyz	Sylhet	Sylhet	207
2	2	abc	xyz	Chittagong	Chittagong	210
3	2	tile	desc	Syl	Syl	123

## 10. All booking show:

All the bookings can be shown.

```
SQL> @"C:\Users\Azmain\OneDrive\Documents\VS Code\DDS Lab\Airbnb\view_1.sql"
View created.
```

BK_ID	CUSTOMER_NAME	CUSTOMER_EMAIL	PRP_T	PRICE_PER_NIGHT	PROPE
1	A	a@gmail.com	abc	1200	POwnB
16-AUG-23	18-JUN-23	2500	done	done	
2	A	a@gmail.com	abc	1000	POwnB
10-SEP-23	13-SEP-23	2700	Unpaid	Active	
3	A	a@gmail.com	abc	1000	POwnB
10-SEP-23	13-SEP-23	2700	Unpaid	Active	

  

BK_ID	CUSTOMER_NAME	CUSTOMER_EMAIL	PRP_T	PRICE_PER_NIGHT	PROPE
4	A	a@gmail.com	abc	1000	POwnB
10-SEP-23	13-SEP-23	2700	Unpaid	Active	
5	A	a@gmail.com	abc	1000	POwnB
10-SEP-23	13-SEP-23	2700	Unpaid	Active	

## 11. 5 most expensive Properties:

Most expensive properties can be seen.

```
SQL> @"C:\Users\Azmain\OneDrive\Documents\VS Code\DDS Lab\Airbnb\view_2.sql"
View created.
```

PRP_ID	PRP_T	PRICE_PER_NIGHT	MAX_G	PROPE
2	abc	1200	2	POwnB
1	abc	1000	5	POwnB

## 12. 10 most booked properties:

Most popular properties can be seen.

```
SQL> @"C:\Users\Azmain\OneDrive\Documents\VS Code\DDS Lab\Airbnb\view_3.sql"
View created.
```

PRP_ID	PRP_T	PRICE_PER_NIGHT	MAX_G	PROPE	TOTAL_BOOKINGS
1	abc	1000	5	POwnB	4
2	abc	1200	2	POwnB	1

## Contribution:

### 1. 190204032:

1. Customer's register and login
2. Book
3. Check personal booking histor
4. Search for property

5. 5 most expensive Properties
  6. 10 most booked properties
2. 190204044:
1. Admin login
  2. Customer password update
  3. Admin Property Add
  4. Payment
  5. All booking show
  6. Booking cancel