University of Petroleum and Energy Studies **Course Code: CSEG1032**

VEHICLE RENTAL SYSTEM

Major Project

C programming

Instructor: Dr. Tanu Singh

Submitted By:

Harsh Raj

SAP ID : 590025597

B.Tech – CSE

ABSTRACT

The Vehicle Rental System is a console based program written in C that makes renting bike, scooty , car easier.

The system stores data like customer name, licence ID, vehicle type, and rental duration and calculating rental costs.

File management, arrays, structures, functions are all used in this project.

By verifying inputs and storing data in a binary file (vehRent.dat) With the addition of features like payment mode integration, billing history,bike availability , DL check and authentication, this application can be expanded into a real-world rental system.

3. Problem Definition

In many small vehicle rental , records are maintained manually which has many difficulty like :

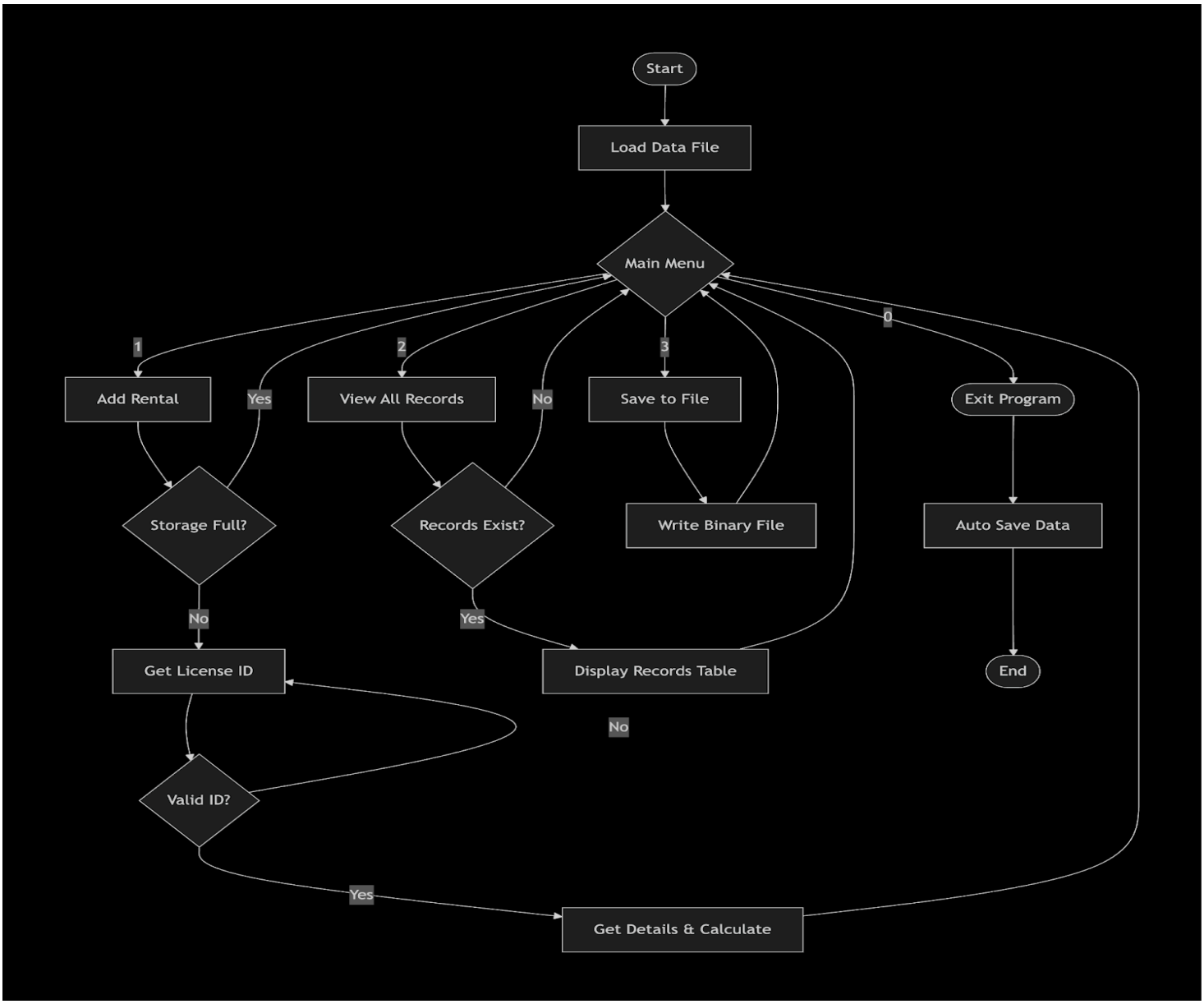
Human errors
Missing entries
Problem in finding customer history
Cost calculation
No data storage in digital format

Therefore, there is a need for a simple, digital rental management system that:

- ✔ **store customer details**
- ✔ **Store rental data**
- ✔ **record of all rentals**
- ✔ **Calculates rental cost**

The objective of this project to create a simple and user-friendly Vehicle Rental System using C.

4. System Design & flowchart



Algorithm for Main

1. Start
2. Load data from file
3. Repeat until user select Exit:
   a. Display menu options
   b. Accept user choice

- If choice = 1, call addRent()
- If choice = 2, call viewAll()
- If choice = 3, call saveFile()
- If choice = 0, save file and exit loop

4. End

ALGORITHM FOR ADDING RENTAL

1. Start
2. If you select rental
3. Input license ID
4. Validate license ID
   - If invalid, repeat step 3
5. Input name
6. Display list of vehicle
7. Input vehicle type
8. If vehicle type is invalid
   - stop
9. Input number of hours
10. Compute cost
    - hours * vrates[vtype]
11. Store rental data
12. Increase rental count
13. Display cost
14. End

ALGORITHM FOR RECORD PRINTING .

1. Start
2. If record count is zero:
   - Display "No Records"
   - Stop
3. Display header
4. For each record :

   Print ID, name, vehicle name, hours, and cost
5. End

ALGORITHM FOR DATA STORING

1. Start
2. Open file "vehRent.dat" in binary write mode
3. Write rental record array to file
4. Close file
5. End

Algorithm for Loading Data

1. Start
2. Open file “vehRent.dat” in binary read mode
3. If file exists:
   - Read entire file into rental array
   - Update record count
4. Close file
5. End

CODE SNIPPETS

- **Structure Definition**

```
struct rentalRec {
    int rid;
    char cname[60];
    int vtype;
    int hrs;
    float cost;
};
```

- **Vehicle Rate List using enum**

```
float vrates[] = {0,50,40,20,70,100,175,200,250,125};
```

```
    r.cost = r.hrs * vrates[r.vtype];
    rentArr[rcount++] = r;
```

Rental Function snippets

```
void addRent()
{
    struct rentalRec r;

    do {
        printf("Enter license ID (5 digits): ");
        scanf("%d", &r.rid);
    } while (r.rid < 10000 || r.rid > 99999);

    printf("Enter Customer Name: ");
    scanf(" %59[^\n]", r.cname);

    showVehicles();
    printf("Choose vehicle: ");
    scanf("%d", &r.vtype);

    printf("Hours taken: ");
    scanf("%d", &r.hrs);

printf("Added. Total: Rs %.2f\n", r.cost);
}
```

Saving File

```
void saveFile()
{
    FILE *fp = fopen("vehRent.dat","wb");
    fwrite(rentArr, sizeof(struct rentalRec),rcount, fp);
    fclose(fp);
}
```

TESTING & RESULT

```
*****************************
   --- RENTAL ROADIES ---
*****************************
1. Add Rental
2. View All
3. Save
0. Exit
  Choice:
```

```
*****************************
   --- RENTAL ROADIES ---
*****************************
1. Add Rental
2. View All
3. Save
0. Exit
  Choice: 1

Enter license ID (5 digits only): 24534
Enter Customer Name: Tarun


---- Vehicles ----

1) Bike (50/hr)                 2) Scooty (40/hr)

3) Cycle (20/hr)                4) Electric Bike (70/hr)

5) Mountain Bike (100/hr)       6) Maruti Suzuki (175/hr)

7) Thar (200/hr)                8) Scorpio (250/hr)

9) Bullet (125/hr)

Choose vehicle: 3
Hours taken: 5
Added. Total: Rs 100.00
```

```
harshraj@HARSHs-MacBook-Air c practice % cd "/Users/harshraj/c practice/" && gcc advnce
ID          | Customer                | Vehicle              | Hours   | Cost       |
-------------------------------------------------------------------------------------
23456       | Abhinit                 | Cycle                | 3       | ₹ 60.00    |
12345       | Harsh                   | Thar                 | 8       | ₹ 1600.00  |
24534       | Tarun                   | Cycle                | 5       | ₹ 100.00   |
45674       | Bhavik                  | Bike                 | 3       | ₹ 150.00   |
12341       | HArsh                   | Electric Bike        | 23      | ₹ 1610.00  |
12349       | Hawert                  | Scorpio              | 23      | ₹ 5750.00  |
12346       | qwert                   | Bike                 | 12      | ₹ 600.00   |
34567       | Harsh Raj               | Bike                 | 8       | ₹ 400.00   |
```

Conclusion & Future Work

This project successfully implements a functional vehicle rental system using C programming.
It meets the objectives of storing customer data, calculating charges, displaying past rentals, and saving data from a file.

Future Enhancements

- Admin login system
- Discount and reward system
- Mobile application
- Online booking
- invoice for booking (PDF)

References

- **Let Us C** – Yashavant Kanetkar
- GeeksforGeeks
- Class notes by DR. TANU SINGH