DBMS – Metro Management System

Submitted By:
Vishal J Lodha
PES1UG20CS507
V Semester Section I

Short Description and Scope of the Project

In a developing country like India where many industries are coming up to boast the economic and social cause like Make in India many factories are coming up in the extensions. Now for people to commute in a safe and economic manner, metro rails have been laid down by the government. Metro provides us with easy, smooth and environment friendly manner to travel from one place to another. Metro provides us with 2 types of options one to use a card and one to use tokens.

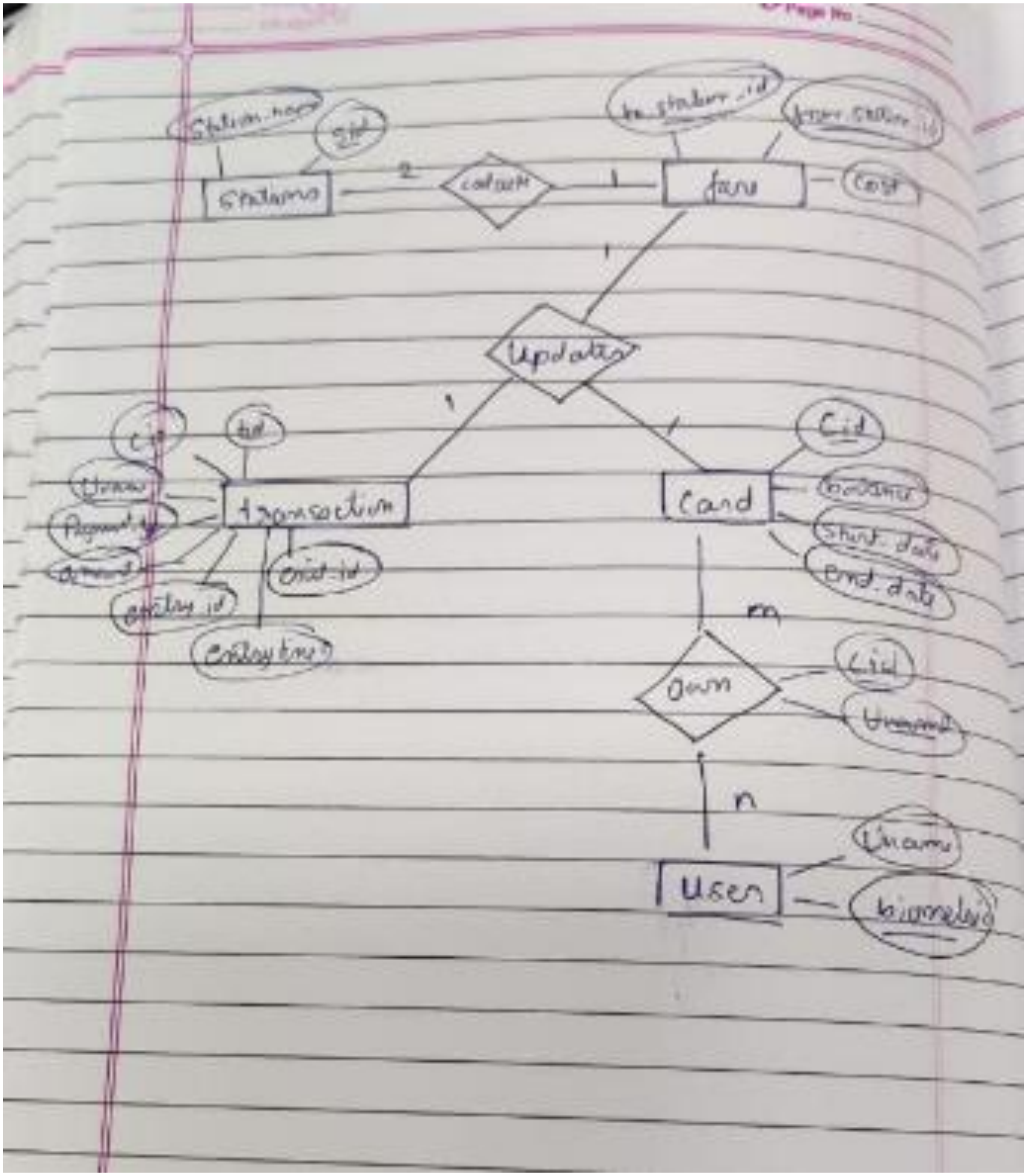
People that use in card tend to have in lots of securities issues such as:

1. Loss of card.
2. Difficulty in keeping track of who is using the card.

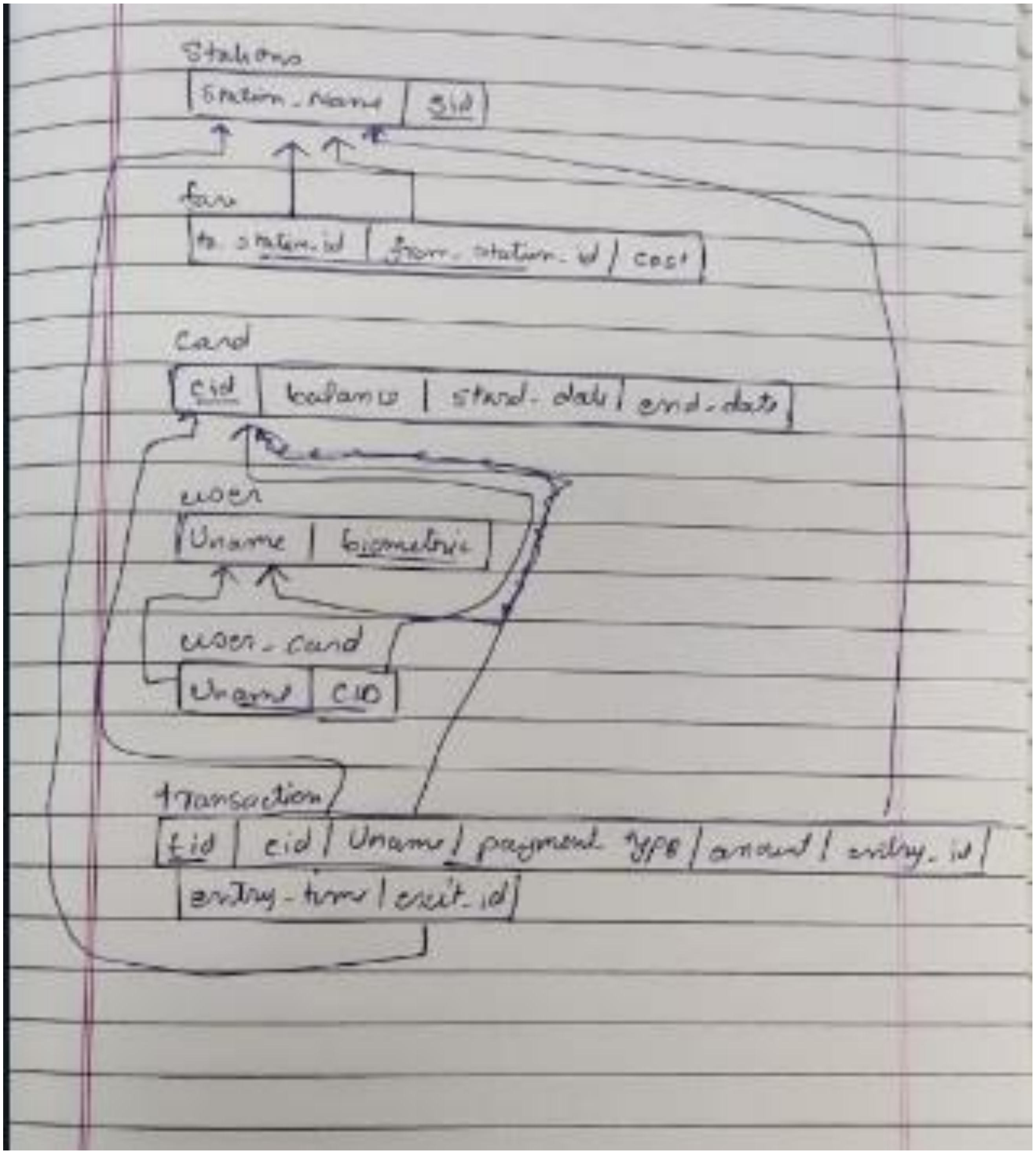
To name a few things.

To overcome this security issues I have created a basic prototype of what I think could be a way to solve in all these issues related to safety and security of the card.

ER Diagram


Station_name
sid
Stations
2
collect
1
to_station_id
from_station_id
fare
cost
1
Updates
1
1
tid
Cid
Uname
transaction
amount
entry_id
exit_id
entry time
Cid
Card
balance
Start_date
End_date
m
Own
Cid
Uname
n
Uname
User

Relational Schema


Stations
Station_Name | sid
fare
to_station_id | from_station_id | cost
Card
cid | balance | start_date | end_date
user
Uname | biometric
user_card
Uname | CID
transaction
tid | cid | Uname | payment type | amount | entry_id
entry_time | exit_id

DDL statements - Building the database

```
    -- createting tables
create table stations(station_name varchar(20) not null,sid int primary key not
null);
create table fare(to_station_id int,from_station_id int,cost
decimal(10,3),foreign key(to_station_id) references stations(sid),foreign
key(from_station_id) references stations(sid),primary
key(to_station_id,from_station_id));
create table card(cid int not null primary key,balance decimal(10,3),start_date
date default curdate(),end_date date);
create table user(Uname varchar(20),biometric binary(5) primary key not null);
create table user_card(Uname varchar(20) not null,CID int not null, foreign
key(CID) references card(cid),primary key(Uname,CID));
create table transaction(tid int not null primary key,cid int not null,Uname
varchar(20),payment_type varchar(10),amount decimal(10,3),entry_id
int,entry_time timestamp,exit_id int,exit_time time,foreign key(cid) references
card(cid),foreign key(entry_id) references stations(sid),foreign key(exit_id)
references stations(sid));
alter table transaction modify column entry_time timestamp default now();
alter table transaction modify column exit_time timestamp default now();
```

Populating the Database

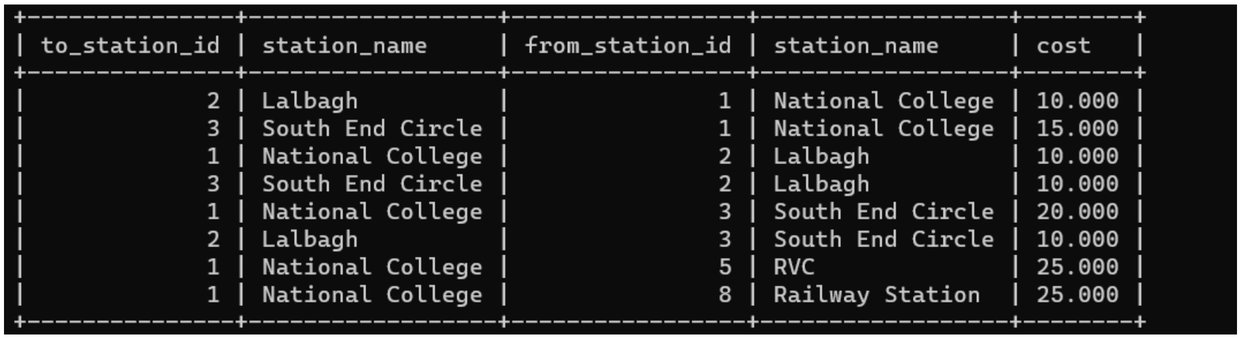
```sql
-- inserting values
 insert into stations values('National Collage',1),('Lalbagh',2),('South End 
Circle',3);
 insert into fare values(1,2,10),(1,3,15),(2,1,10),(2,3,10),(3,1,15),(3,2,10);
 insert into card(cid,balance) values(1234,150.0),(1243,500),(1342,75.42);
 insert into user 
values('Vishal','00110'),('Nitish','11111'),('Yash','11001'),('Yashas','10101');
 insert into user_card 
values('Vishal',1234),('Vishal',1243),('Yash',1243),('Nitish',1342),('Yashas',13
42),('Yashas',1243);
 insert into transaction(tid,cid,Uname,payment_type,amount,entry_id) 
values(1,1234,'Vishal','cash',100,1),(4,1342,'Nitish','online',50,2);
 insert into transaction(tid,cid,Uname,payment_type,amount,entry_id,exit_id) 
values(2,1243,'Yashas','card',-10,2,1),(3,1243,'Vishal','card',-15,1,3);
```

Join Queries

1. To show the Station name,id along with travelling cost

   Query:
   select to_station_id,s.station_name,from_station_id,si.station_name,cost from stations s join fare f on f.to_station_id=s.sid join stations si on f.from_station_id=si.sid;

   Result:

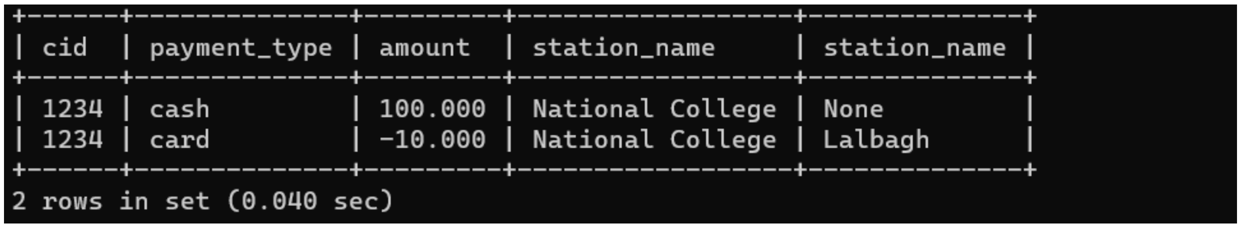


```
+---------------+-----------------+-----------------+-----------------+--------+
| to_station_id | station_name    | from_station_id | station_name    | cost   |
+---------------+-----------------+-----------------+-----------------+--------+
|             2 | Lalbagh         |               1 | National College | 10.000 |
|             3 | South End Circle |               1 | National College | 15.000 |
|             1 | National College |               2 | Lalbagh         | 10.000 |
|             3 | South End Circle |               2 | Lalbagh         | 10.000 |
|             1 | National College |               3 | South End Circle | 20.000 |
|             2 | Lalbagh         |               3 | South End Circle | 10.000 |
|             1 | National College |               5 | RVC             | 25.000 |
|             1 | National College |               8 | Railway Station | 25.000 |
+---------------+-----------------+-----------------+-----------------+--------+
```

2. To show the travel history of a particular person

   Query:
   select cid,payment_type,amount,s.station_name,si.station_name from transaction join stations s on entry_id=s.sid join stations si on exit_id=si.sid where cid in(select cid from user_card where Uname='Vishal');

   Result:

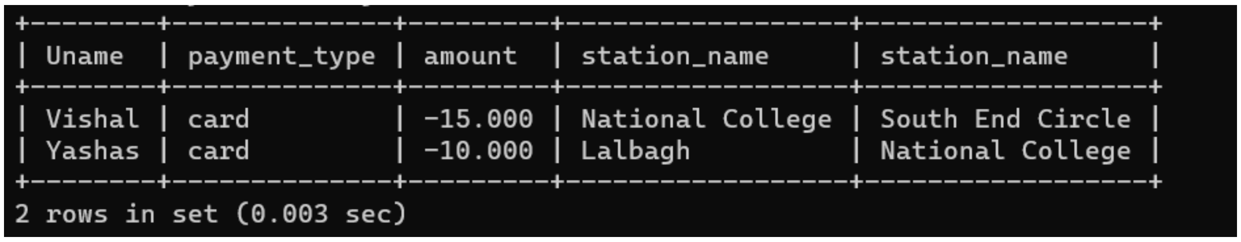


```
+------+--------------+---------+------------------+--------------+
| cid  | payment_type | amount  | station_name     | station_name |
+------+--------------+---------+------------------+--------------+
| 1234 | cash         | 100.000 | National College | None         |
| 1234 | card         | -10.000 | National College | Lalbagh      |
+------+--------------+---------+------------------+--------------+
2 rows in set (0.040 sec)
```

3. To show the travel history related to a particular card

Query:
select Uname,payment_type,amount,s.station_name,si.station_name from transaction join stations s on entry_id=s.sid join stations si on exit_id=si.sid where cid in(select cid from user_card where cid=1243);

Result:

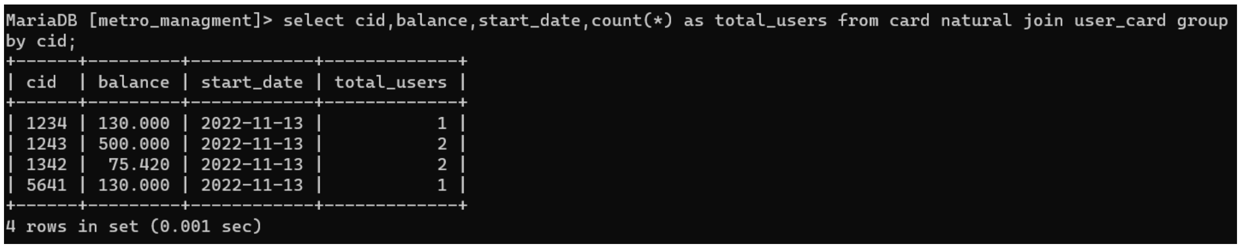


```
+--------+--------------+---------+-----------------+-----------------+
| Uname  | payment_type | amount  | station_name    | station_name    |
+--------+--------------+---------+-----------------+-----------------+
| Vishal | card         | -15.000 | National College | South End Circle |
| Yashas | card         | -10.000 | Lalbagh         | National College |
+--------+--------------+---------+-----------------+-----------------+
2 rows in set (0.003 sec)
```

4. To find no of user and card id, balance for all cards

   Query:
   select cid,balance,start_date,count(*) as total_users from card natural join user_card group by cid;

   Result:



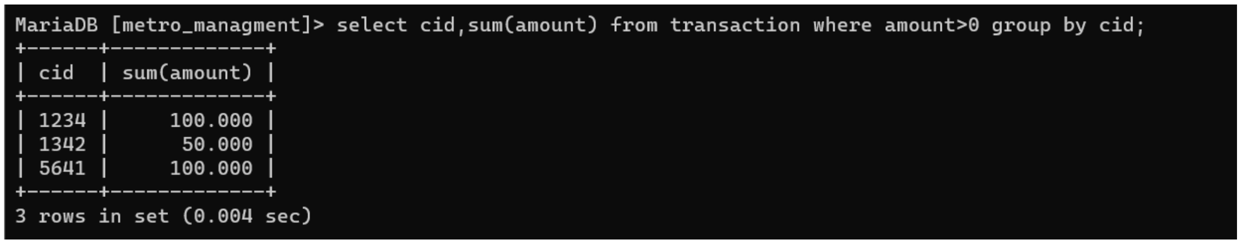
```
MariaDB [metro_managment]> select cid,balance,start_date,count(*) as total_users from card natural join user_card group
by cid;
+------+---------+------------+-------------+
| cid  | balance | start_date | total_users |
+------+---------+------------+-------------+
| 1234 | 130.000 | 2022-11-13 |           1 |
| 1243 | 500.000 | 2022-11-13 |           2 |
| 1342 |  75.420 | 2022-11-13 |           2 |
| 5641 | 130.000 | 2022-11-13 |           1 |
+------+---------+------------+-------------+
4 rows in set (0.001 sec)
```

Aggregate Functions

1. Total top up for each card

   Query:
   select cid,sum(amount) from transaction where amount>0 group by cid;

   Result:

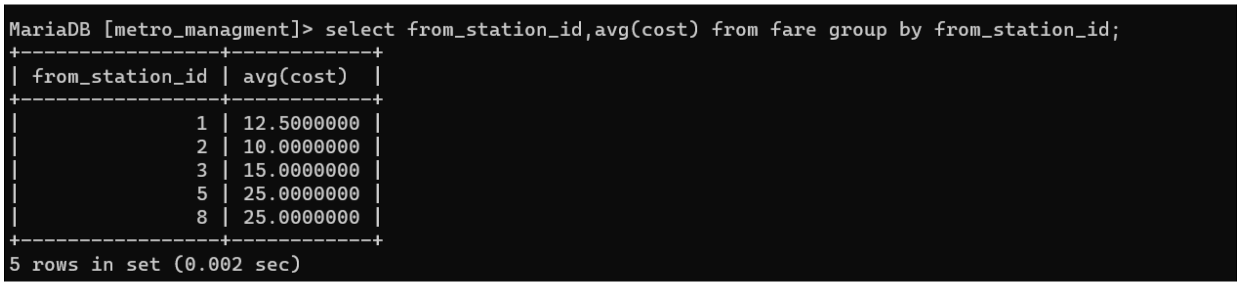


```
MariaDB [metro_managment]> select cid,sum(amount) from transaction where amount>0 group by cid;
+------+-------------+
| cid  | sum(amount) |
+------+-------------+
| 1234 |     100.000 |
| 1342 |      50.000 |
| 5641 |     100.000 |
+------+-------------+
3 rows in set (0.004 sec)
```

2. Find average cost for a travel

Query:
select from_station_id,avg(cost) from fare group by from_station_id;

Result:

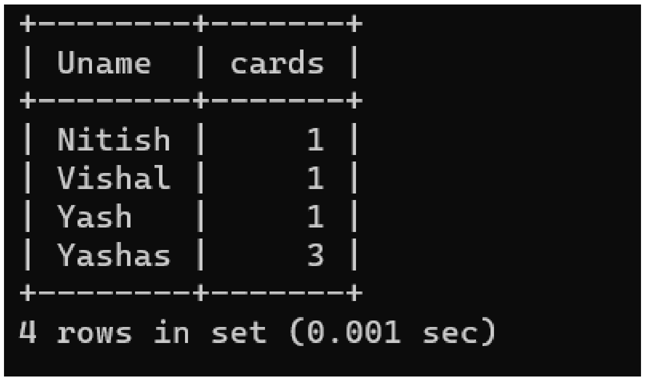


```
MariaDB [metro_managment]> select from_station_id,avg(cost) from fare group by from_station_id;
+-----------------+------------+
| from_station_id | avg(cost)  |
+-----------------+------------+
|               1 | 12.5000000 |
|               2 | 10.0000000 |
|               3 | 15.0000000 |
|               5 | 25.0000000 |
|               8 | 25.0000000 |
+-----------------+------------+
5 rows in set (0.002 sec)
```

3. Total cards owned by users.

Query:
select Uname,count(*) as cards from user_card group by Uname;

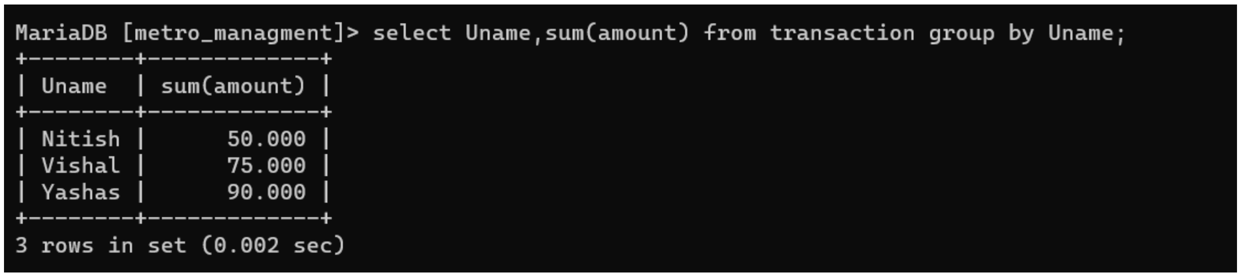
Result:


```
+--------+-------+
| Uname  | cards |
+--------+-------+
| Nitish |     1 |
| Vishal |     1 |
| Yash   |     1 |
| Yashas |     3 |
+--------+-------+
4 rows in set (0.001 sec)
```

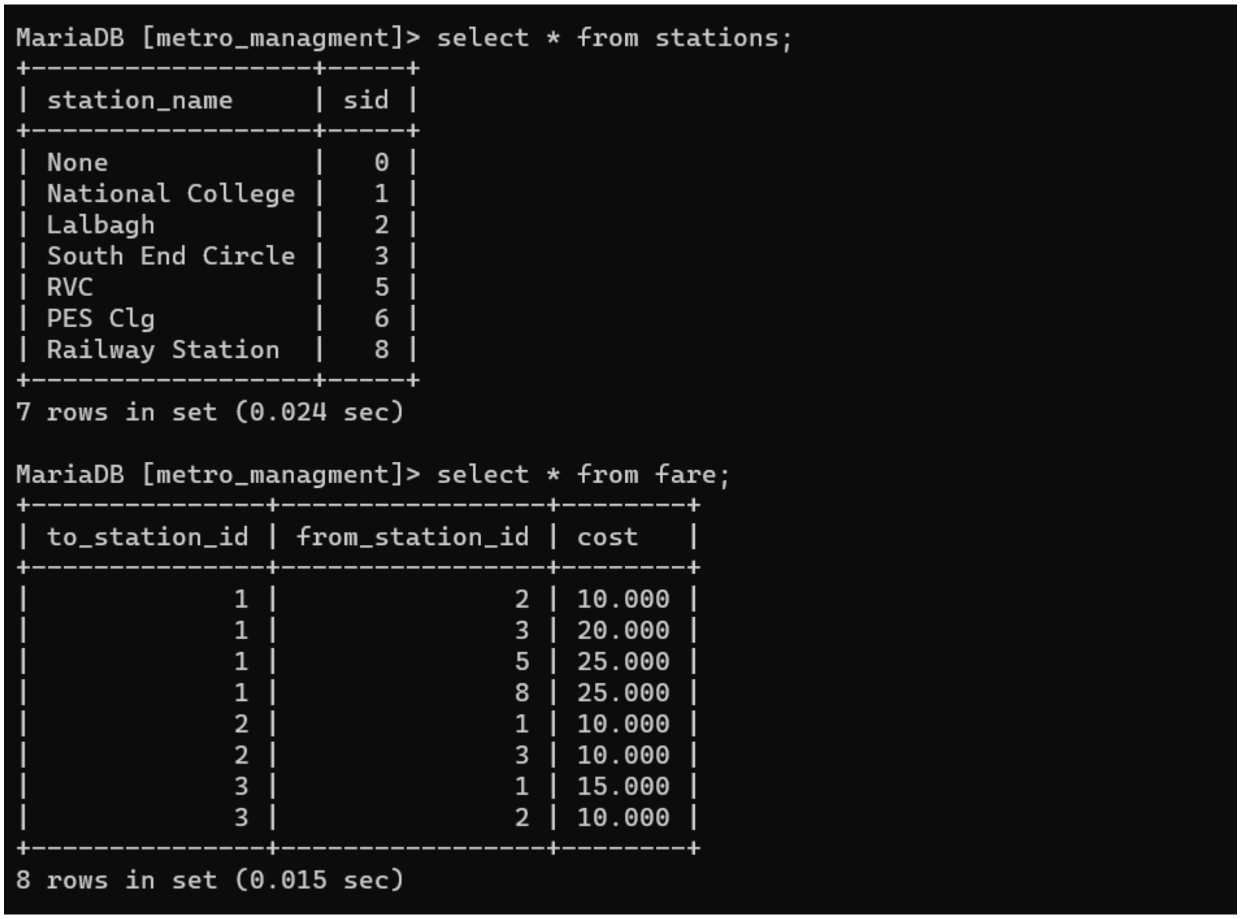
4. Total transactions per user

Query:
select Uname,sum(amount) from transaction group by Uname;

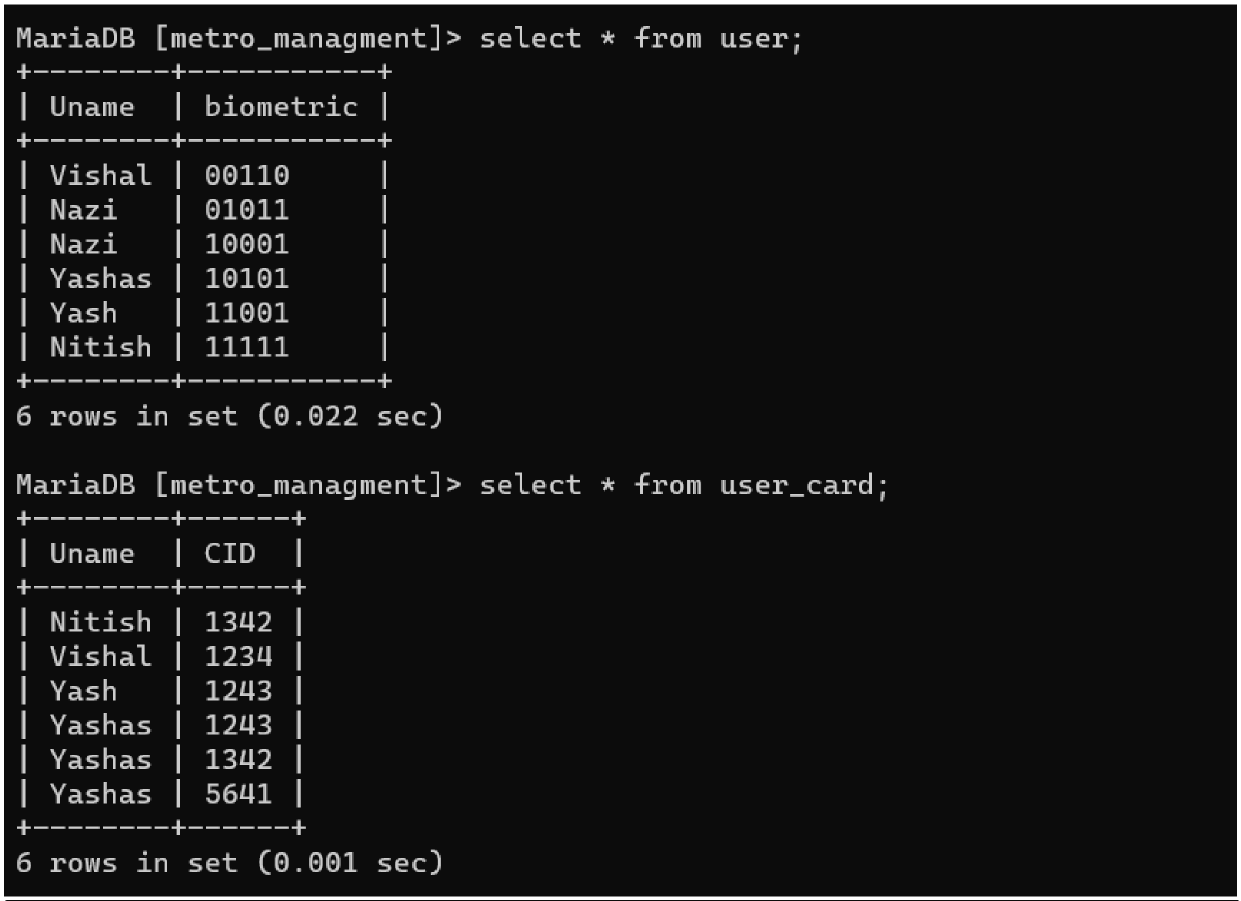
Result:


```
MariaDB [metro_managment]> select Uname,sum(amount) from transaction group by Uname;
+--------+-------------+
| Uname  | sum(amount) |
+--------+-------------+
| Nitish |      50.000 |
| Vishal |      75.000 |
| Yashas |      90.000 |
+--------+-------------+
3 rows in set (0.002 sec)
```

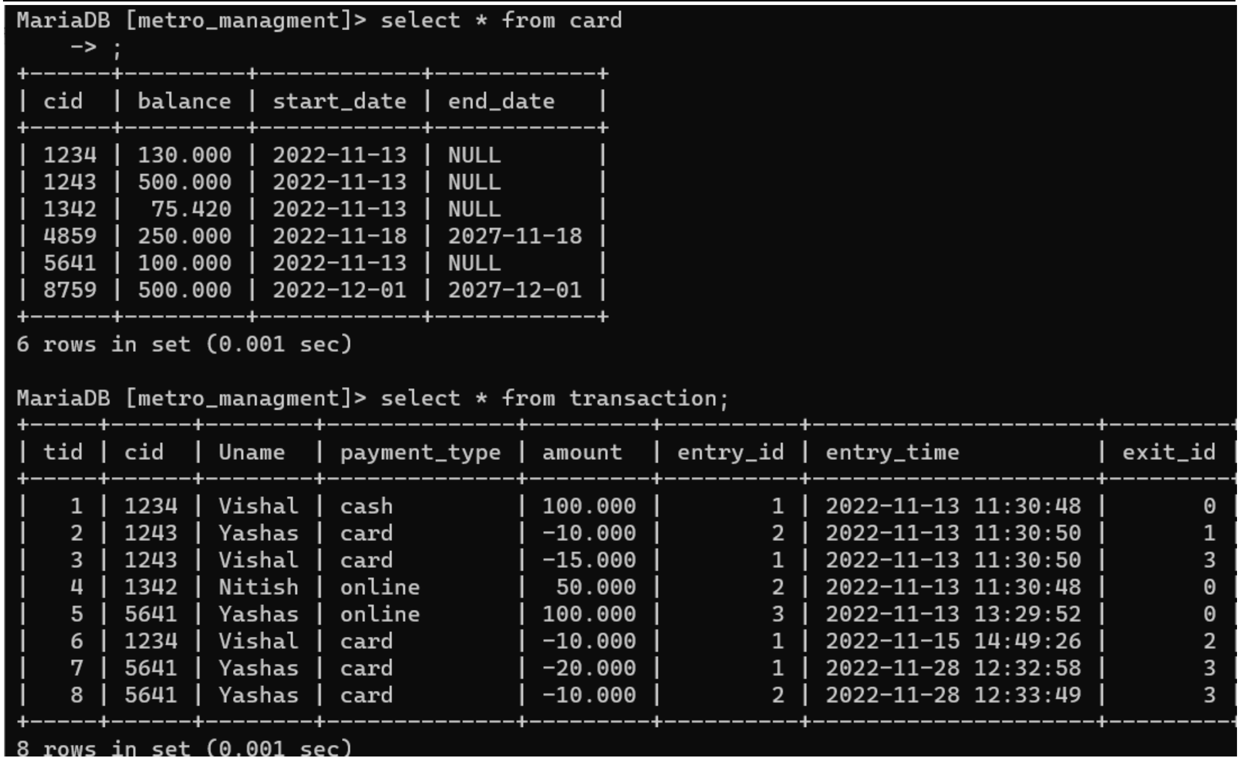
Tables



```
MariaDB [metro_managment]> select * from stations;
+-----------------+-----+
| station_name    | sid |
+-----------------+-----+
| None            |   0 |
| National College |   1 |
| Lalbagh         |   2 |
| South End Circle |   3 |
| RVC             |   5 |
| PES Clg         |   6 |
| Railway Station |   8 |
+-----------------+-----+
7 rows in set (0.024 sec)

MariaDB [metro_managment]> select * from fare;
+---------------+-----------------+--------+
| to_station_id | from_station_id | cost   |
+---------------+-----------------+--------+
|             1 |               2 | 10.000 |
|             1 |               3 | 20.000 |
|             1 |               5 | 25.000 |
|             1 |               8 | 25.000 |
|             2 |               1 | 10.000 |
|             2 |               3 | 10.000 |
|             3 |               1 | 15.000 |
|             3 |               2 | 10.000 |
+---------------+-----------------+--------+
8 rows in set (0.015 sec)
```



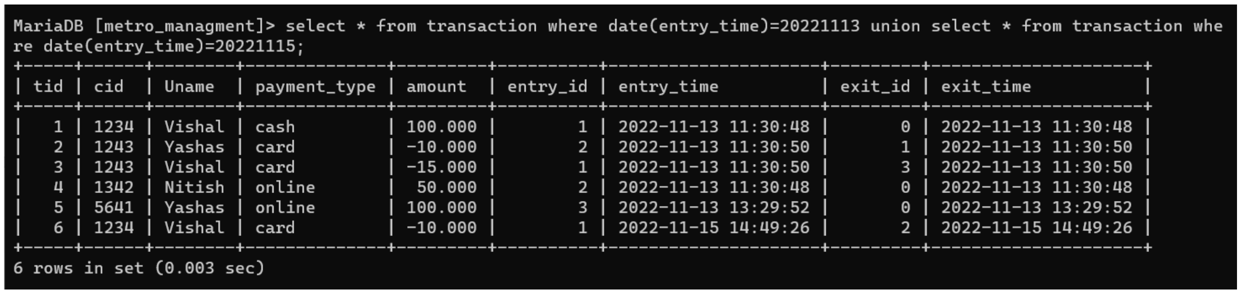
```
MariaDB [metro_managment]> select * from user;
+--------+-----------+
| Uname  | biometric |
+--------+-----------+
| Vishal | 00110     |
| Nazi   | 01011     |
| Nazi   | 10001     |
| Yashas | 10101     |
| Yash   | 11001     |
| Nitish | 11111     |
+--------+-----------+
6 rows in set (0.022 sec)

MariaDB [metro_managment]> select * from user_card;
+--------+------+
| Uname  | CID  |
+--------+------+
| Nitish | 1342 |
| Vishal | 1234 |
| Yash   | 1243 |
| Yashas | 1243 |
| Yashas | 1342 |
| Yashas | 5641 |
+--------+------+
6 rows in set (0.001 sec)
```



```
MariaDB [metro_managment]> select * from card
    -> ;
+------+---------+------------+------------+
| cid  | balance | start_date | end_date   |
+------+---------+------------+------------+
| 1234 | 130.000 | 2022-11-13 | NULL       |
| 1243 | 500.000 | 2022-11-13 | NULL       |
| 1342 |  75.420 | 2022-11-13 | NULL       |
| 4859 | 250.000 | 2022-11-18 | 2027-11-18 |
| 5641 | 100.000 | 2022-11-13 | NULL       |
| 8759 | 500.000 | 2022-12-01 | 2027-12-01 |
+------+---------+------------+------------+
6 rows in set (0.001 sec)

MariaDB [metro_managment]> select * from transaction;
+-----+------+--------+--------------+---------+----------+---------------------+---------
| tid | cid  | Uname  | payment_type | amount  | entry_id | entry_time          | exit_id |
+-----+------+--------+--------------+---------+----------+---------------------+---------
|   1 | 1234 | Vishal | cash         | 100.000 |        1 | 2022-11-13 11:30:48 |       0 |
|   2 | 1243 | Yashas | card         | -10.000 |        2 | 2022-11-13 11:30:50 |       1 |
|   3 | 1243 | Vishal | card         | -15.000 |        1 | 2022-11-13 11:30:50 |       3 |
|   4 | 1342 | Nitish | online       |  50.000 |        2 | 2022-11-13 11:30:48 |       0 |
|   5 | 5641 | Yashas | online       | 100.000 |        3 | 2022-11-13 13:29:52 |       0 |
|   6 | 1234 | Vishal | card         | -10.000 |        1 | 2022-11-15 14:49:26 |       2 |
|   7 | 5641 | Yashas | card         | -20.000 |        1 | 2022-11-28 12:32:58 |       3 |
|   8 | 5641 | Yashas | card         | -10.000 |        2 | 2022-11-28 12:33:49 |       3 |
+-----+------+--------+--------------+---------+----------+---------------------+---------
8 rows in set (0.001 sec)
```

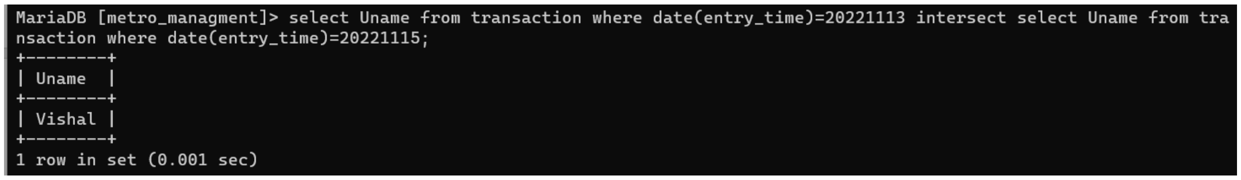
Set Operations

1)People travelling on 2022/11/13 and 2022/11/15



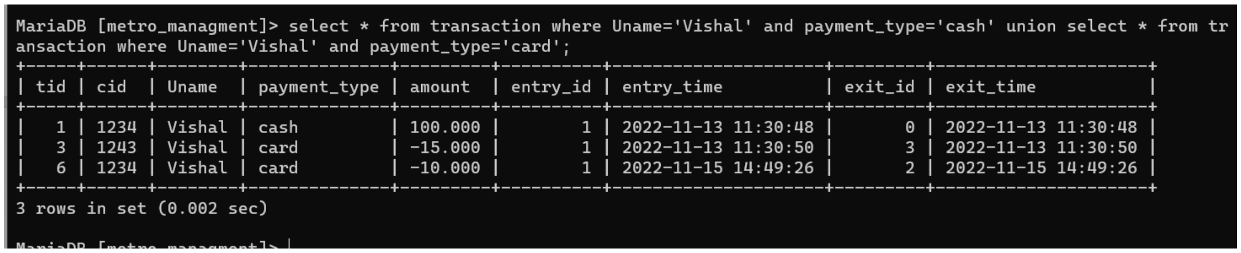
```
MariaDB [metro_managment]> select * from transaction where date(entry_time)=20221113 union select * from transaction whe
re date(entry_time)=20221115;
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
| tid | cid  | Uname  | payment_type | amount  | entry_id | entry_time          | exit_id | exit_time           |
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
|   1 | 1234 | Vishal | cash         | 100.000 |        1 | 2022-11-13 11:30:48 |       0 | 2022-11-13 11:30:48 |
|   2 | 1243 | Yashas | card         | -10.000 |        2 | 2022-11-13 11:30:50 |       1 | 2022-11-13 11:30:50 |
|   3 | 1243 | Vishal | card         | -15.000 |        1 | 2022-11-13 11:30:50 |       3 | 2022-11-13 11:30:50 |
|   4 | 1342 | Nitish | online       |  50.000 |        2 | 2022-11-13 11:30:48 |       0 | 2022-11-13 11:30:48 |
|   5 | 5641 | Yashas | online       | 100.000 |        3 | 2022-11-13 13:29:52 |       0 | 2022-11-13 13:29:52 |
|   6 | 1234 | Vishal | card         | -10.000 |        1 | 2022-11-15 14:49:26 |       2 | 2022-11-15 14:49:26 |
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
6 rows in set (0.003 sec)
```

2) Common user travelling on both days



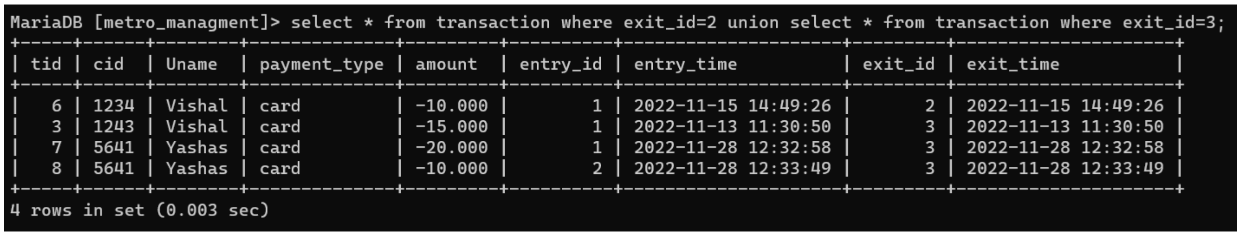
```
MariaDB [metro_managment]> select Uname from transaction where date(entry_time)=20221113 intersect select Uname from tra
nsaction where date(entry_time)=20221115;
+--------+
| Uname  |
+--------+
| Vishal |
+--------+
1 row in set (0.001 sec)
```

3) All cash and card payments by the user Vishal



```
MariaDB [metro_managment]> select * from transaction where Uname='Vishal' and payment_type='cash' union select * from tr
ansaction where Uname='Vishal' and payment_type='card';
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
| tid | cid  | Uname  | payment_type | amount  | entry_id | entry_time          | exit_id | exit_time           |
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
|   1 | 1234 | Vishal | cash         | 100.000 |        1 | 2022-11-13 11:30:48 |       0 | 2022-11-13 11:30:48 |
|   3 | 1243 | Vishal | card         | -15.000 |        1 | 2022-11-13 11:30:50 |       3 | 2022-11-13 11:30:50 |
|   6 | 1234 | Vishal | card         | -10.000 |        1 | 2022-11-15 14:49:26 |       2 | 2022-11-15 14:49:26 |
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
3 rows in set (0.002 sec)

MariaDB [metro_managment]>
```

4) All transaction history of users exiting at station id 3 or station id 2



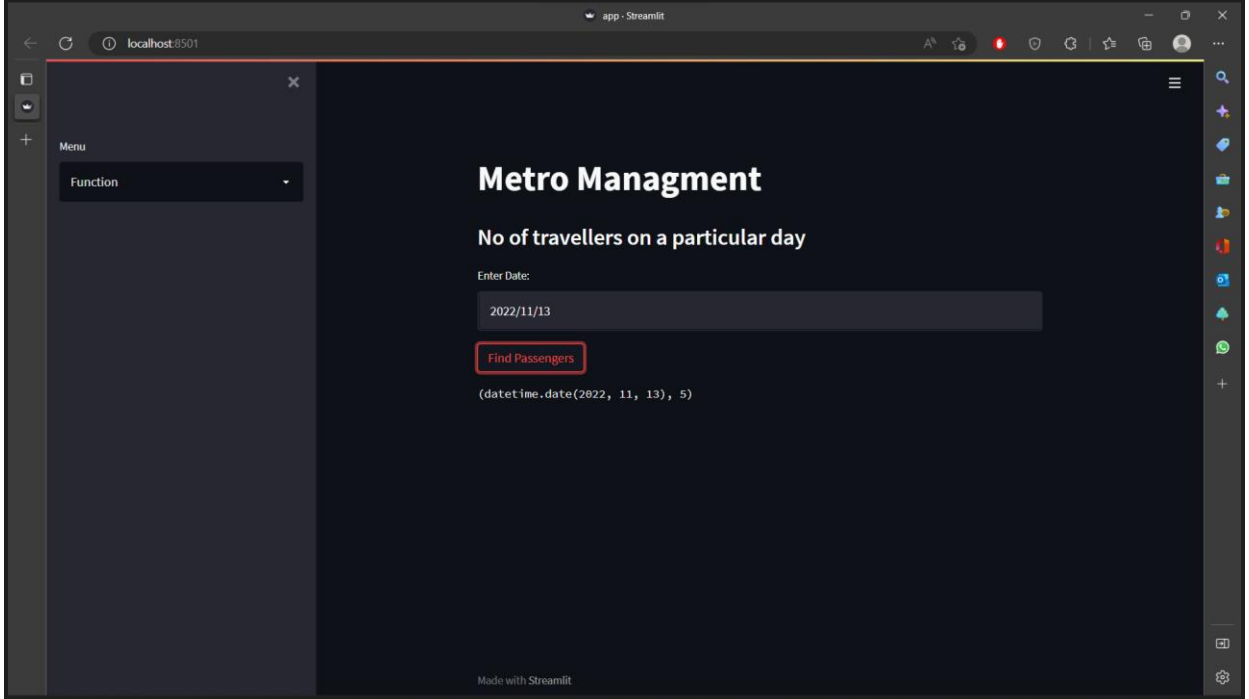
```
MariaDB [metro_managment]> select * from transaction where exit_id=2 union select * from transaction where exit_id=3;
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
| tid | cid  | Uname  | payment_type | amount  | entry_id | entry_time          | exit_id | exit_time           |
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
|   6 | 1234 | Vishal | card         | -10.000 |        1 | 2022-11-15 14:49:26 |       2 | 2022-11-15 14:49:26 |
|   3 | 1243 | Vishal | card         | -15.000 |        1 | 2022-11-13 11:30:50 |       3 | 2022-11-13 11:30:50 |
|   7 | 5641 | Yashas | card         | -20.000 |        1 | 2022-11-28 12:32:58 |       3 | 2022-11-28 12:32:58 |
|   8 | 5641 | Yashas | card         | -10.000 |        2 | 2022-11-28 12:33:49 |       3 | 2022-11-28 12:33:49 |
+-----+------+--------+--------------+---------+----------+---------------------+---------+---------------------+
4 rows in set (0.003 sec)
```

Functions and Procedures

Fucntion:

```
function no of passengers travelling on a specific day
```

```
delimiter $$
create function passen(tdate date)
returns int
deterministic
begin
declare ret int;
select count(*) into ret from transaction where date(entry_time)=tdate and
exit_id<>0;
return ret;
end $$
delimiter
```



Metro Managment

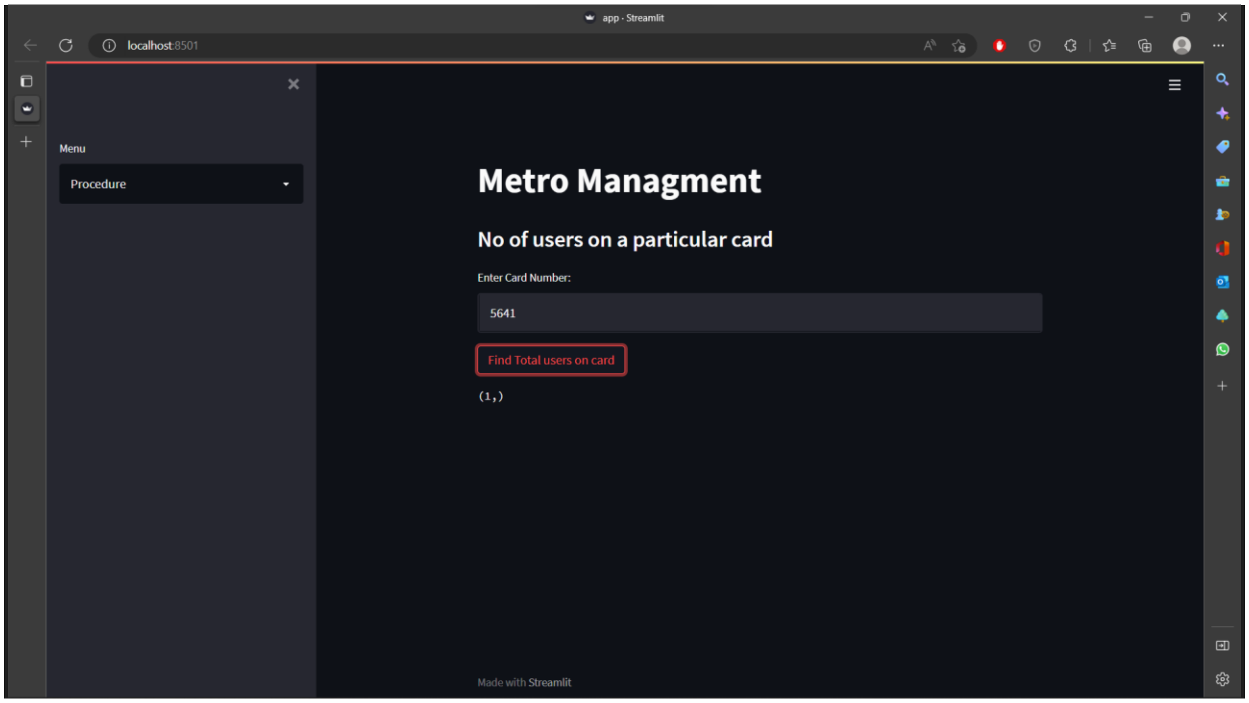
No of travellers on a particular day

Enter Date:

2022/11/13

```
(datetime.date(2022, 11, 13), 5)
```

```
Procedure:
procedure to return the no of users using a particular card

delimiter $$
create procedure card_users(in carid int)
deterministic
begin
select count(*) as users from user_card where CID=carid;
end $$
delimiter
```



Metro Managment

No of users on a particular card

Enter Card Number:

5641

Find Total users on card

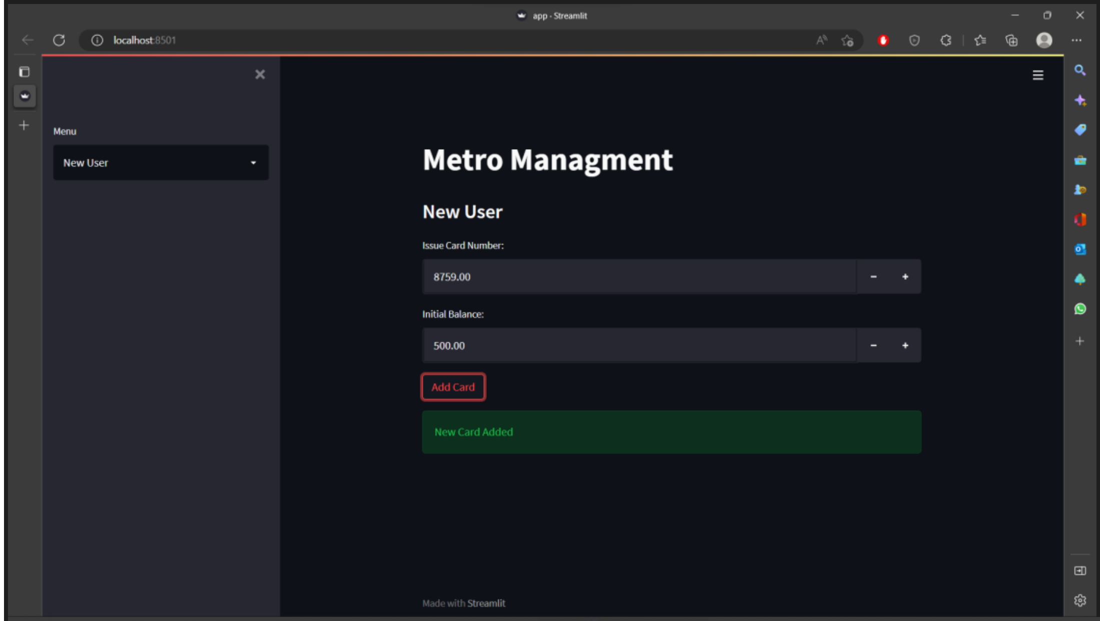
```
(1,)
```

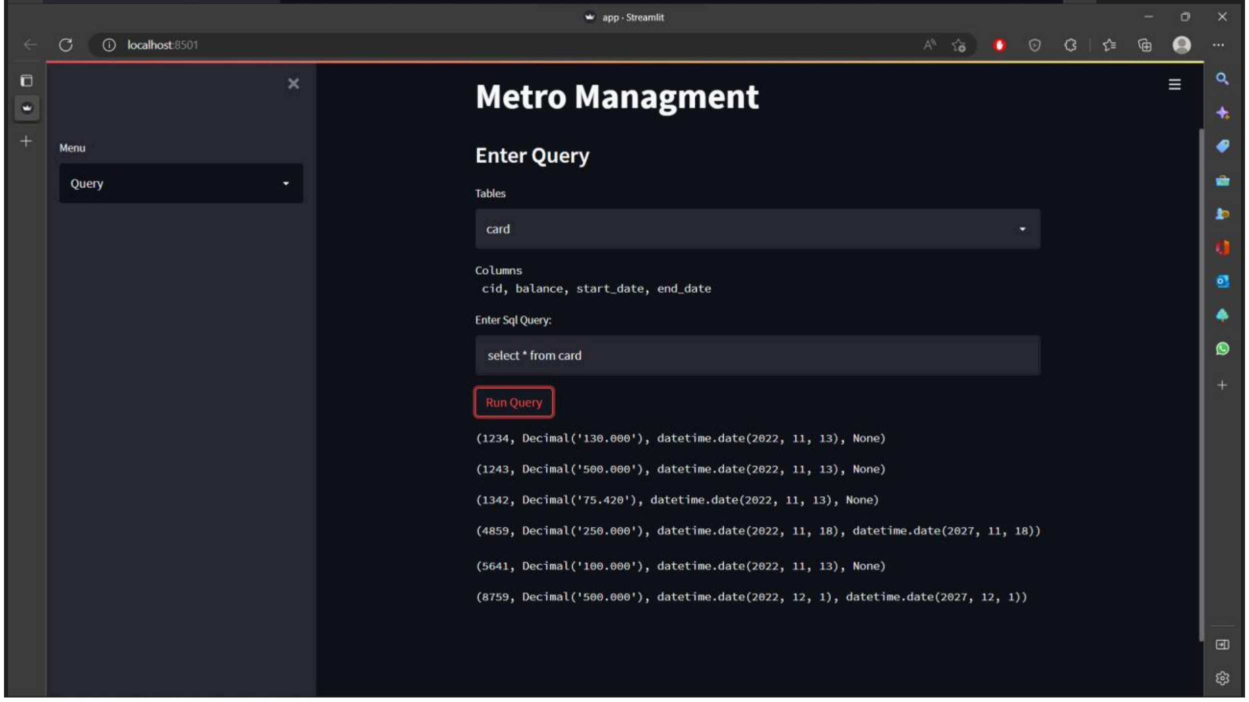
Trigges and cursor

Trigger:

```
triggers to set end_date for a card
```

```
delimiter $$
create trigger exp_dat before insert on card for each row
begin
set new.end_date=new.start_date+00050000;
end $$
delimiter
```



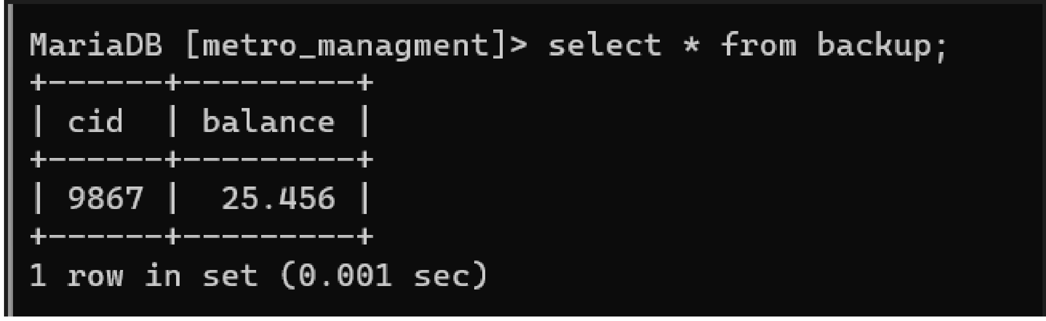


```
Columns
 cid, balance, start_date, end_date
```

```
select * from card
```

```
(1234, Decimal('130.000'), datetime.date(2022, 11, 13), None)

(1243, Decimal('500.000'), datetime.date(2022, 11, 13), None)

(1342, Decimal('75.420'), datetime.date(2022, 11, 13), None)

(4859, Decimal('250.000'), datetime.date(2022, 11, 18), datetime.date(2027, 11, 18))

(5641, Decimal('100.000'), datetime.date(2022, 11, 13), None)

(8759, Decimal('500.000'), datetime.date(2022, 12, 1), datetime.date(2027, 12, 1))
```

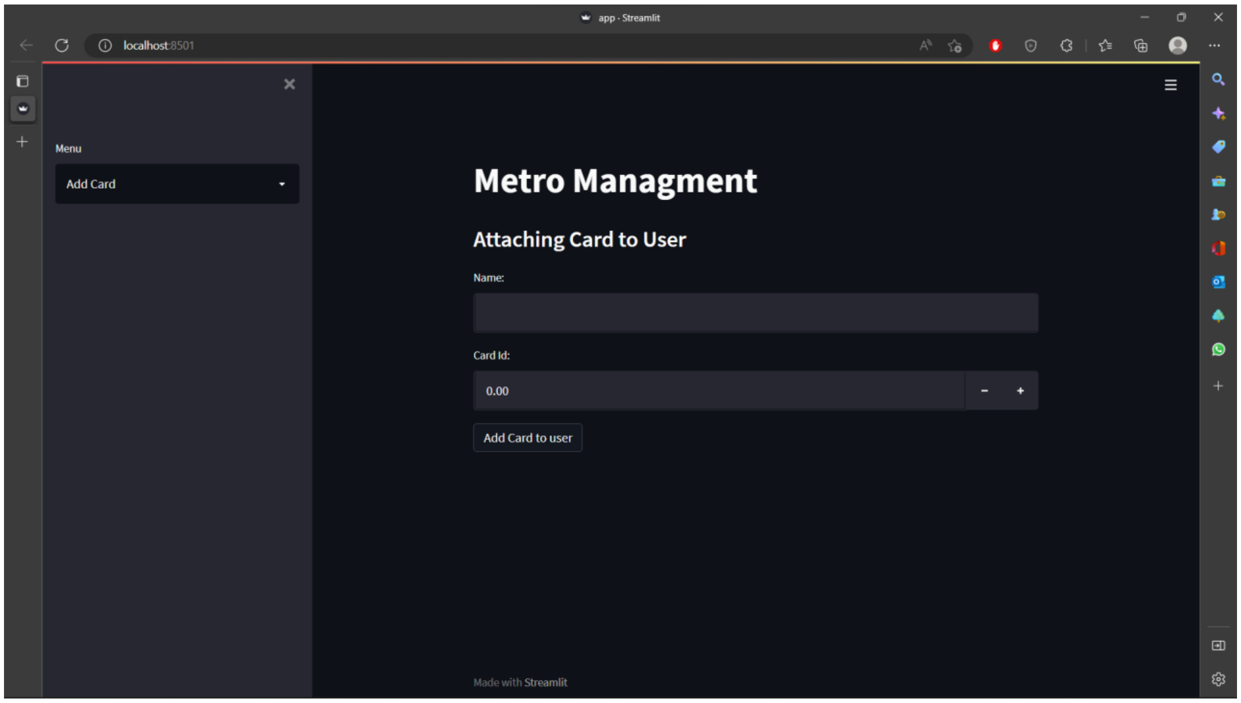
```
Cursor:
Create backup table which contains expiry cards
delimitier $$
create procedure bup()
deterministic
begin
declare done int default 0;
declare c int;
declare b decimal(10,3);
declare e date;
declare cur cursor for select cid,balance,end_date from card;
declare continue handler for not found set done=1;
open cur;
label:Loop
fetch cur into c,b,e;
if e>curdate() THEN
insert into backup values(c,e);
end if;
if done=1 then leave label;
end if;
end loop;
close cur;
end $$
delimiter;
```



```
MariaDB [metro_managment]> select * from backup;
+------+---------+
| cid  | balance |
+------+---------+
| 9867 |  25.456 |
+------+---------+
1 row in set (0.001 sec)
```

Developing a Frontend

Attaching card to a specific user



Metro Managment

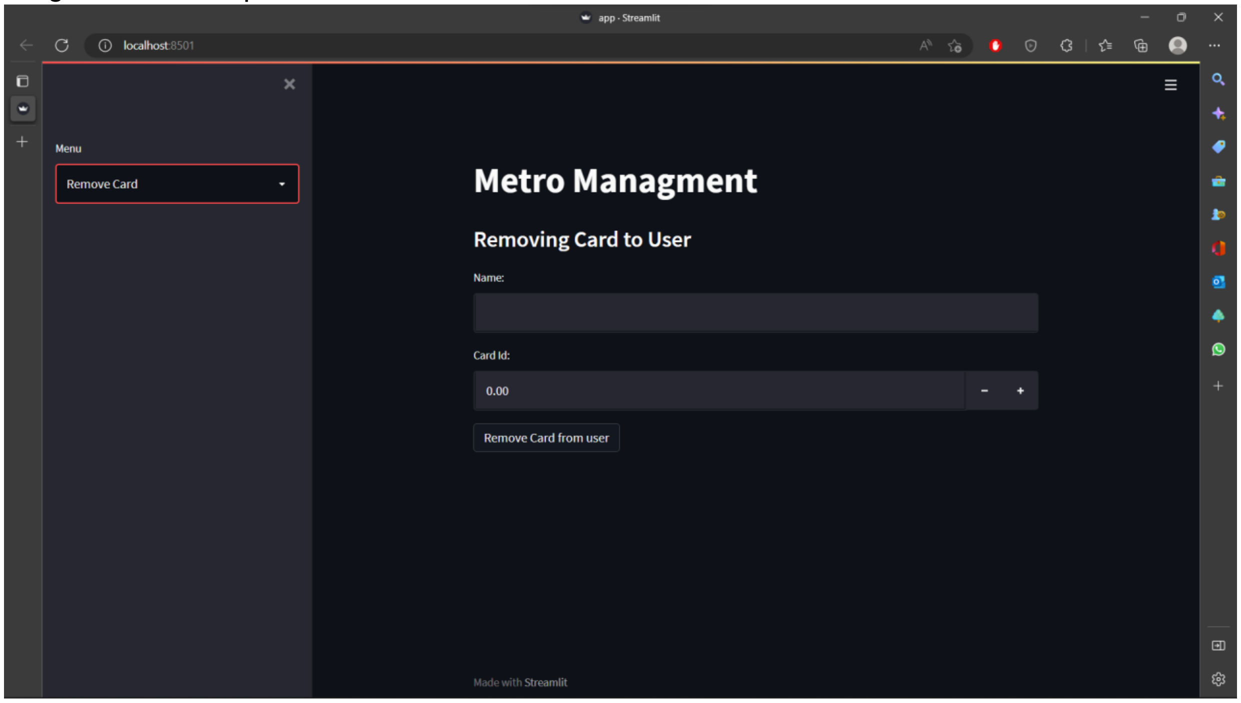
Attaching Card to User

Name:

Card Id:

0.00

Removing card from a specific user



Metro Managment

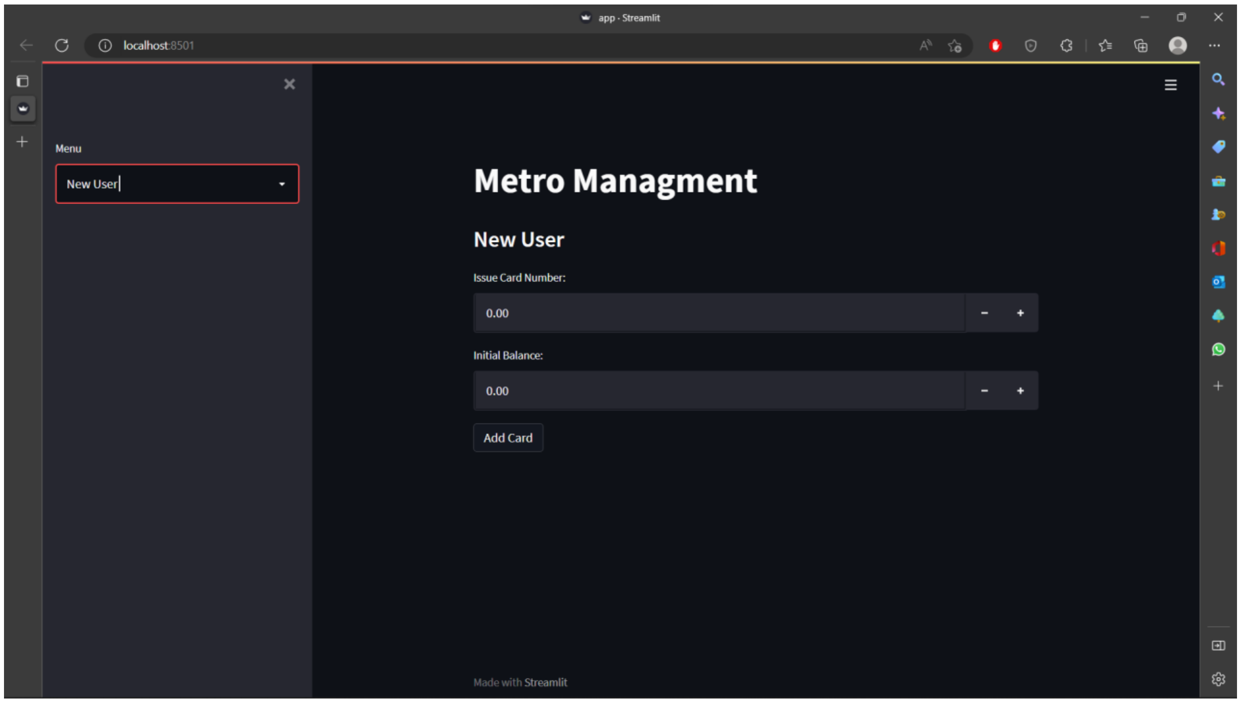
Removing Card to User

Name:

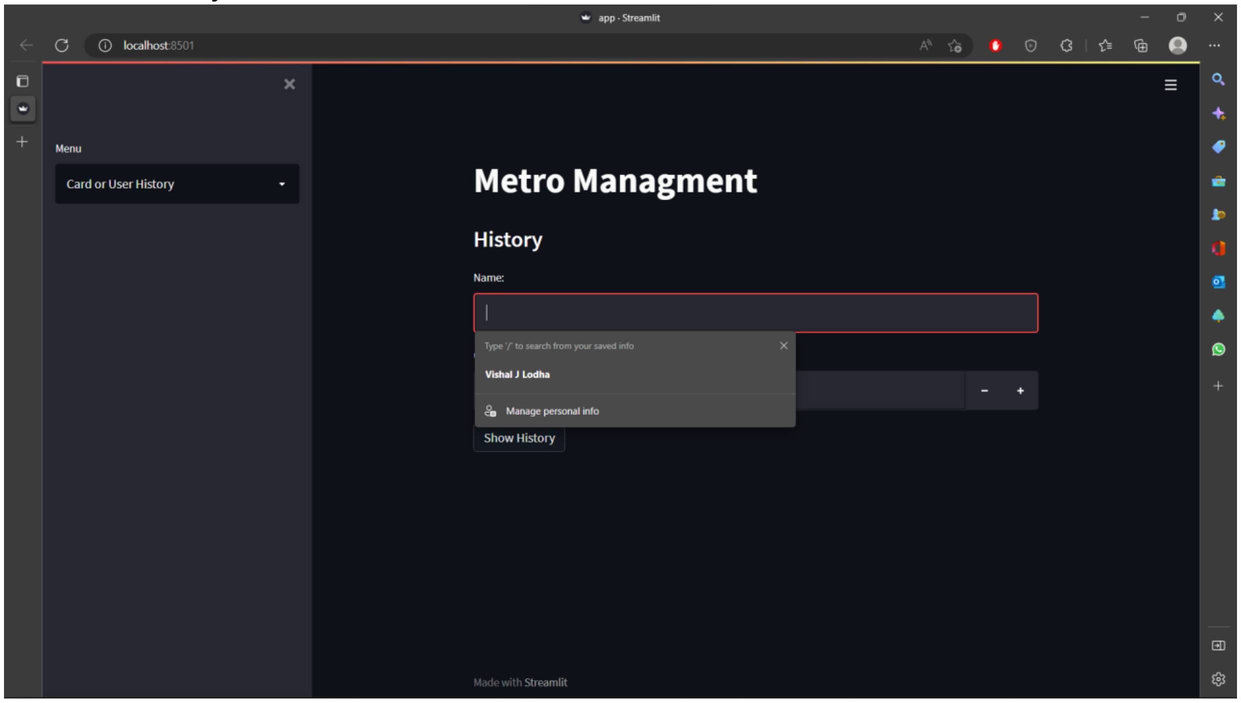
Card Id:

0.00

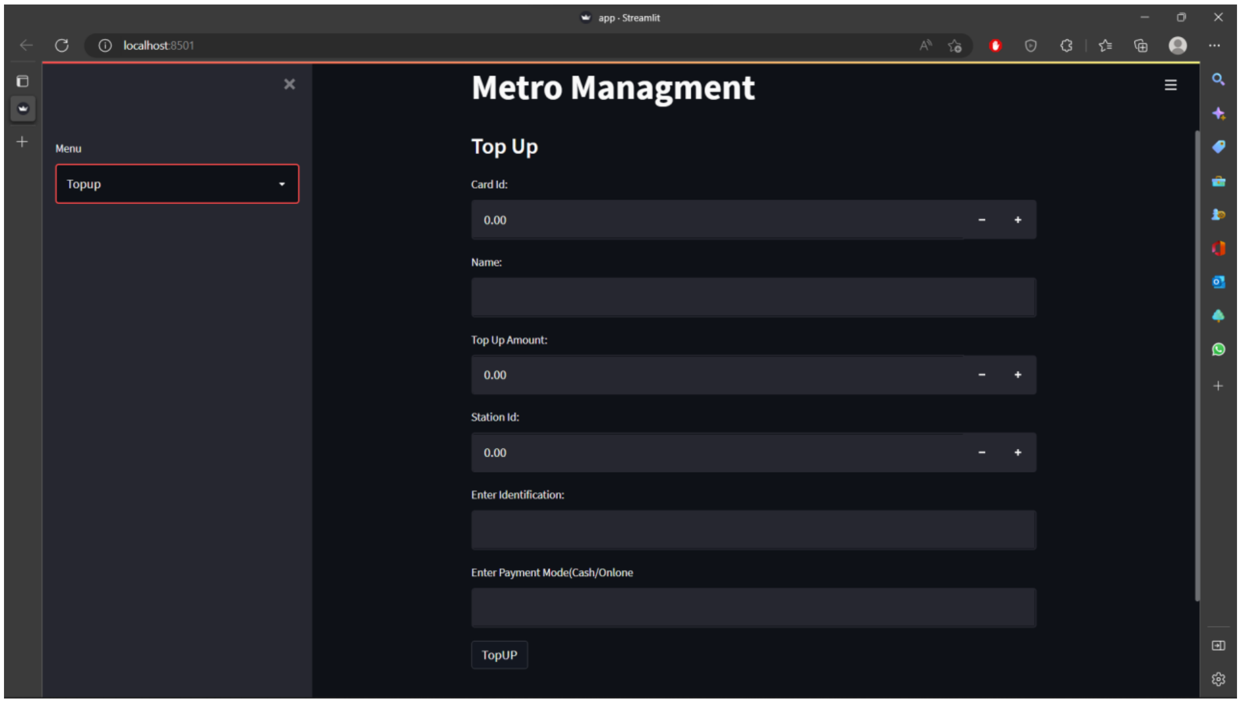
Issuing new card



User card history



Top up



Metro Managment

Top Up

Menu: Topup

Card Id: 0.00

Name:

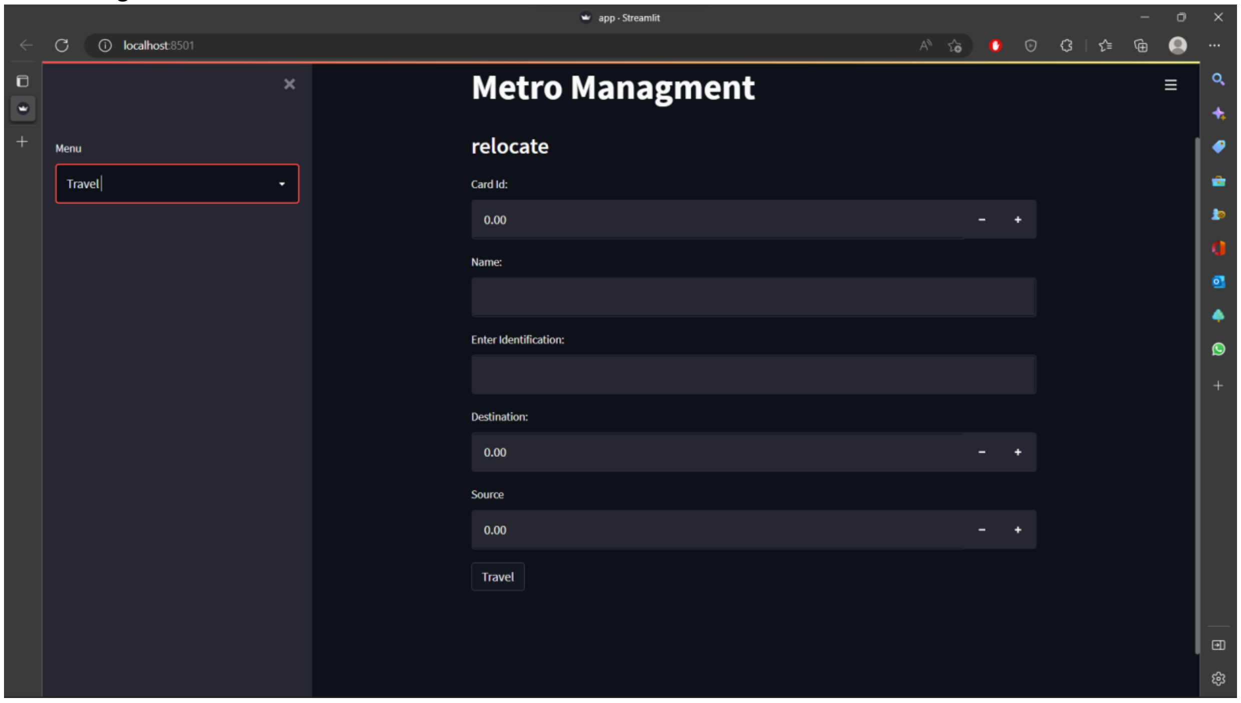
Top Up Amount: 0.00

Station Id: 0.00

Enter Identification:

Enter Payment Mode(Cash/Onlone

Travelling



Metro Managment

relocate

Menu: Travel

Card Id: 0.00

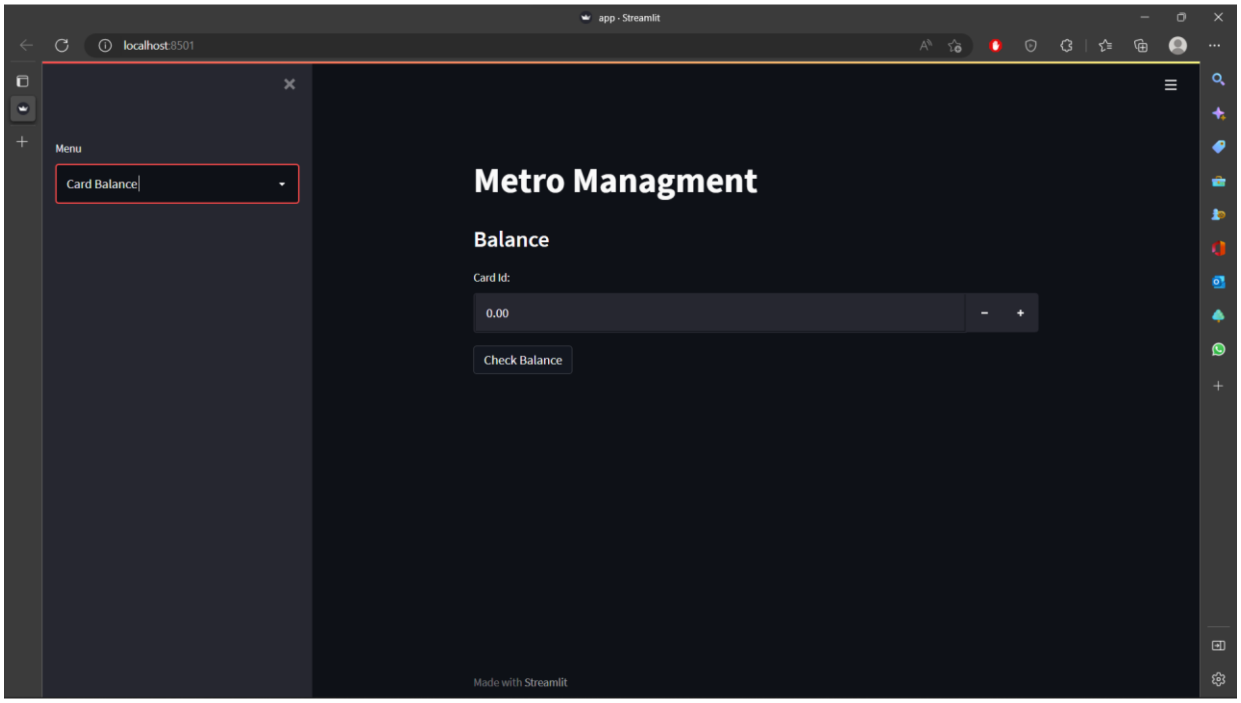
Name:

Enter Identification:

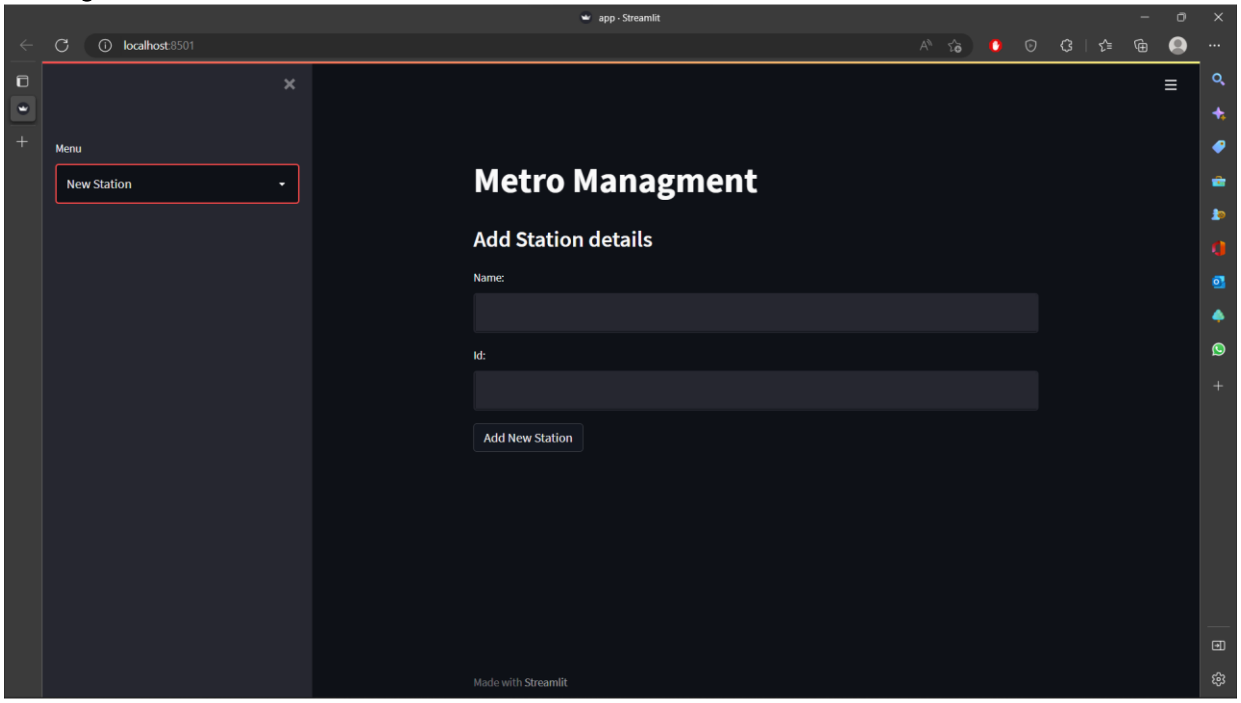
Destination: 0.00

Source 0.00

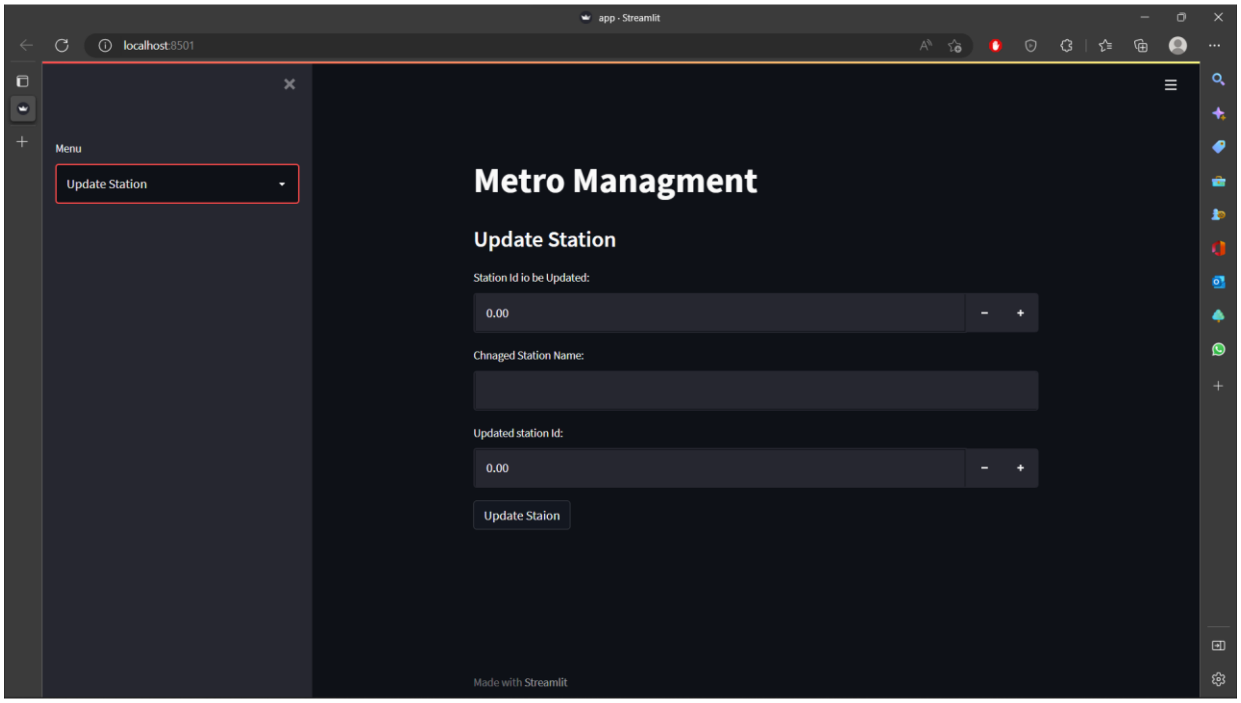
Checking balance of a specific card



Adding new Station to the metro line



Updating station details



Metro Managment

Update Station

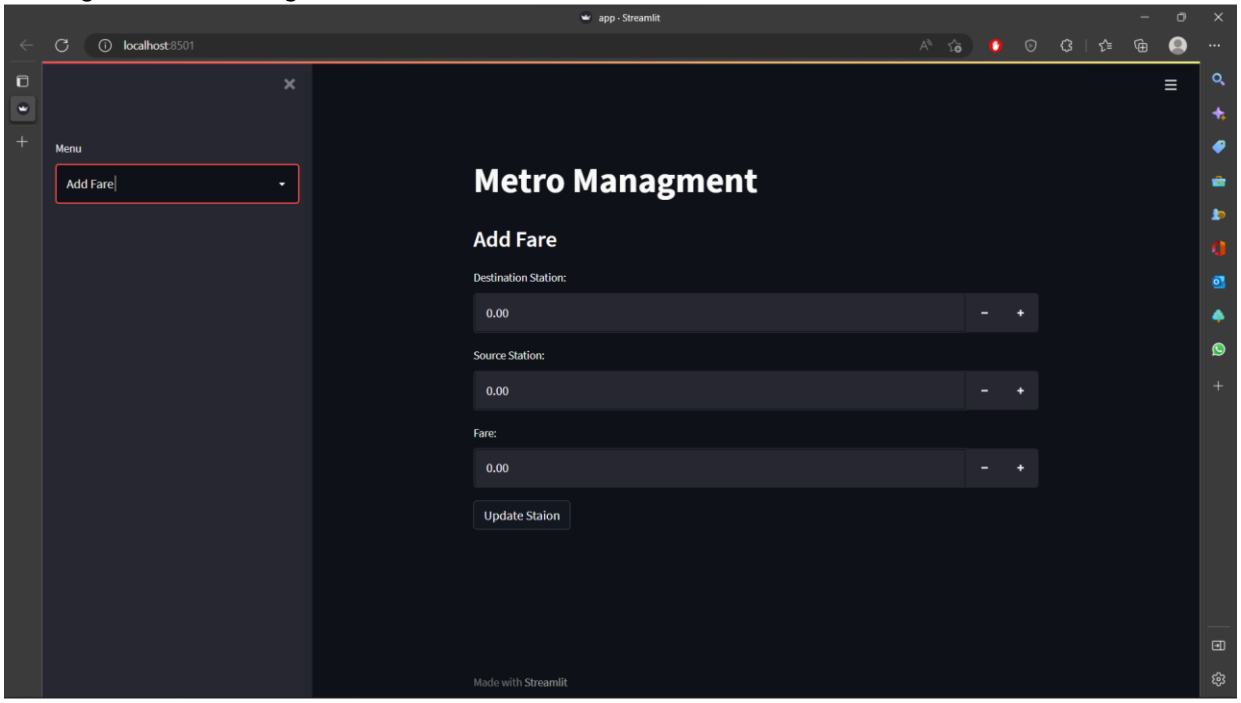
Station Id io be Updated:
0.00

Chnaged Station Name:

Updated station Id:
0.00

Update Staion

Adding fare for moving from onr station to another



Metro Managment

Add Fare

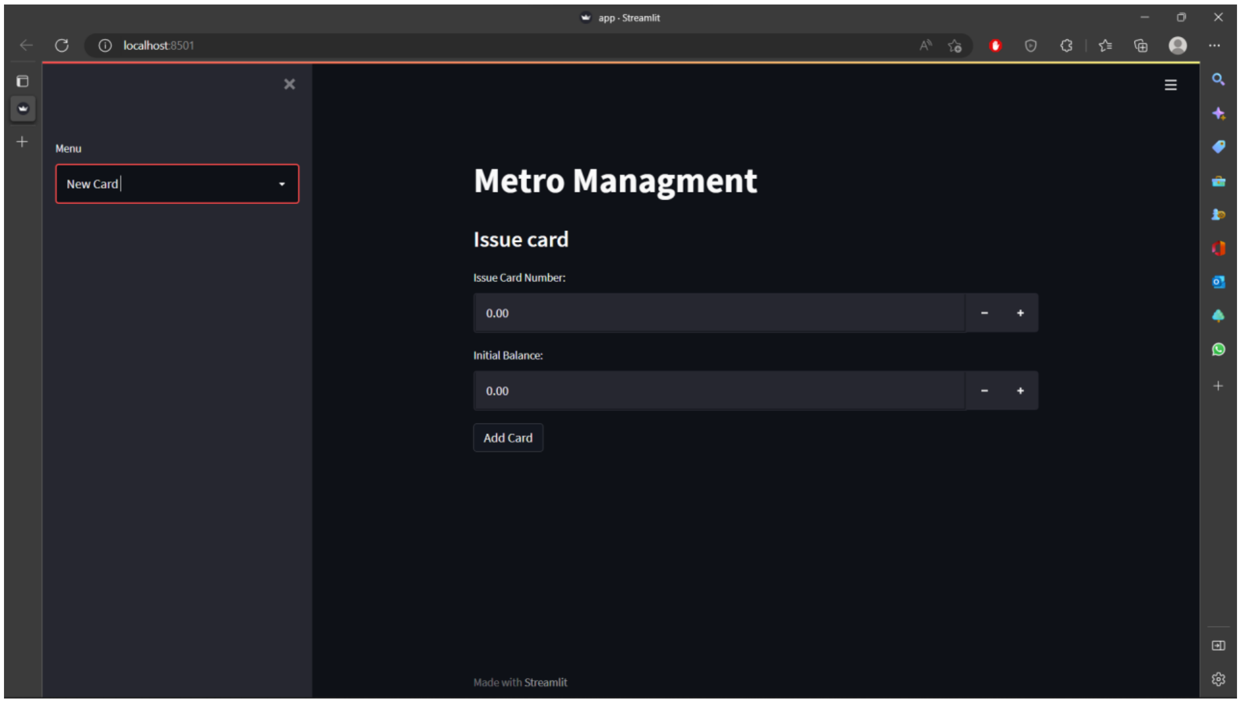
Destination Station:
0.00

Source Station:
0.00

Fare:
0.00

Update Staion

Issue card



Metro Managment

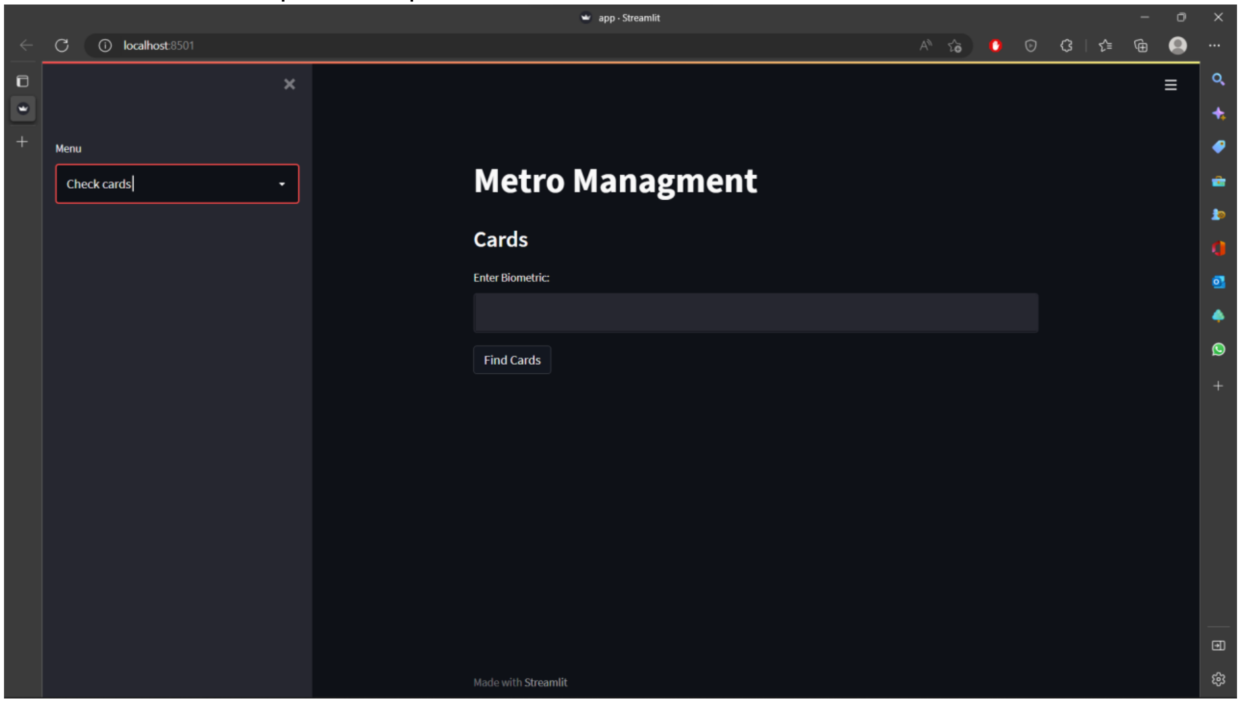
Issue card

Issue Card Number: 0.00

Initial Balance: 0.00

Add Card

Check all cards for a particular person



Metro Managment

Cards

Enter Biometric:

Find Cards