

Documentation of DBMS Project

Group:-Nikhil kumar(2017301)

Deepak Kumar Thakur(2017337)

login.java-

checking username and password from database whether it exist or not, if exist then match all the details of that user.

```
1)Select * from login where UserName="'+user.getText()+'" and  
Password="'+pass.getText().toString()+'";
```

Registerlogin.java-

Getting usernames details from login and if it not already exist then register with confirm password

```
1)select UserName from login;  
2)insert into login(UserName>Password)values(?,?);
```

UserDetails.java-

Checking usernames from login users for duplicity and if not duplicates found then add, if found then show already exist error message and then insert into login table and then fill permanent details in user table as this is important for user to book tickets because this contains account balance and other important details

```
1) select UserName from login;  
2) insert into login(UserName>Password)values(?,?);  
3) insert into  
user(First_Name>Last_Name,Adhaar_No,Gender,Age,Mobile,EmailID>Password,City,State,Us  
erName)values(?,?,?,?,?,?,?,?,?,?);
```

SeatsForm.java-

Firstly get the available seats from train table with tno that we have from booktickets and then we get the details of train for required train and print it in table then getting wallet values to show available balance in your account and checking for insufficient balance if required amount for book tickets from range 1 to 4 and then if amount <=balance then book tickets to confirm payment then it gives the payment details to PaymentHistory.java and update the wallet balance and train available tickets accordingly

```
1) Select seats_Avail from train where Train_No="'+tno+'";  
2) Select * from train where Train_Name ='+s+'";  
3) Select Wallet from user where UserName="'+login.name+'";
```

```

4)select TicketID from paymentdetails;

5)Update user set Wallet =""+(Integer.parseInt(wallet.getText())-f)+" where
UserName="" +this.st+"";

6)Update train set seats_Avail =""+(seatoriginal-
Integer.parseInt((String)seat.getSelectedItemAt()))+" where Train_No="" +tno+"";

7)insert into
paymentdetails(PayID,TicketID,Train_No,Date,Time,Amount,Balance,UserName,seat_booked)values
(?,?,?,?,?,?,?,?,?);

```

PaymentHistory.java-

```

1)Select PayID,TicketID,Train_No,Date,Time,Amount,Balance from paymentdetails where
UserName="" +login.name+"";

```

SeatAvail.java-

Getting Train name and seats avail from train as we have train number from textbox so just checking is required train with train no.in text box if it exist or not

```

1)Select Train_Name,seats_Avail from train where Train_No = "" +number.getText()+"";

```

CheckAvail.java-

Getting train name, seats_avail and arrive time important details for user to get the information to check if he is available for that time or not and seats are available or not and train name is also equally important

```

1)Select Train_Name,seats_Avail,Arrive_time from train where Train_No = "" +this.num+"";

```

CancelTickets.java-

Getting train_no, seats booked from payment details that is required to get the seats avail from train that we will get from tno that we are getting from paymentdetails and seats_Avail is required to update the train table value of seats_Avail and then getting Username and Amount from payment details to update the values of wallet using Username and Amount and then we require details for cancelled_records then for that we are getting those details from paymentdetails those values are=PayID,Train_No,seat_booked and lastly we delete the required record from payment details after getting all data from all the table which we require.

```

1)Select Train_No,seat_booked from paymentdetails where TicketID="" +ticket.getText()+"";

2)Select seats_Avail from train where Train_No="" +tno+"";

3)Update train set seats_Avail ="" +seat_cancelled+" where Train_No="" +tno+"";

4)Select UserName,Amount from paymentdetails where TicketID="" +ticket.getText()+"";

```

5)Select Wallet from user where UserName='"+login.name+"';

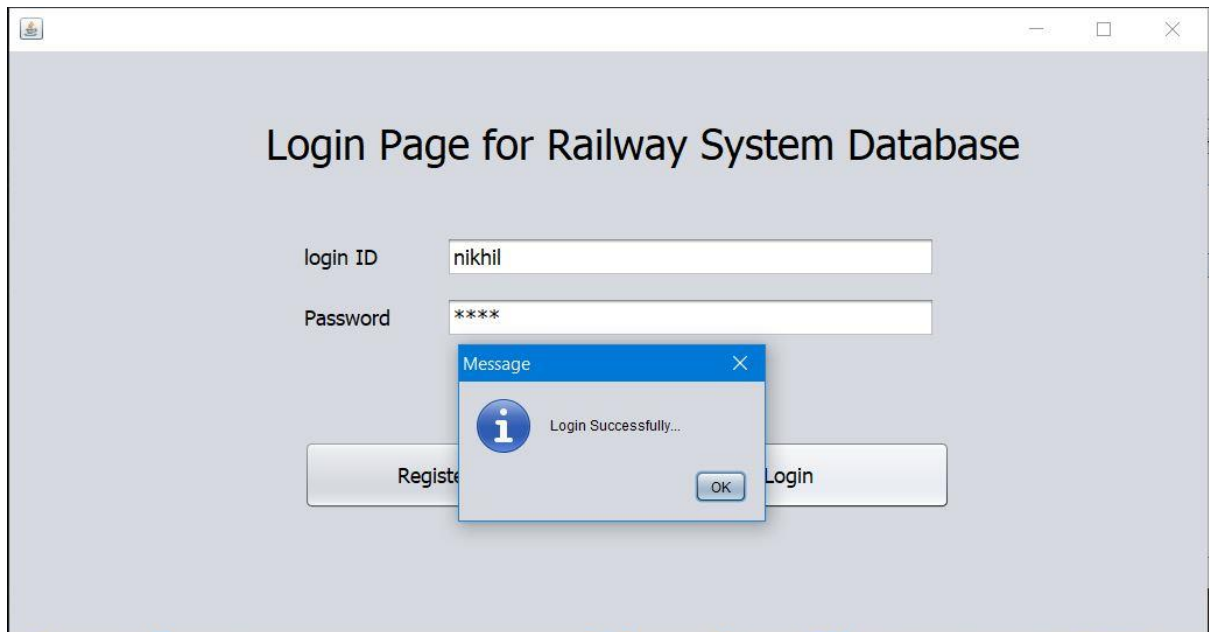
6)Update user set Wallet='"+(amount+wallet_balance_of_user)+"' where
UserName='"+user+"';

7)Select PayID,Train_No,seat_booked from paymentdetails where
TicketID='"+ticket.getText()+"';

8)Insert into
cancelled_records(PayID,TicketID,Train_No,Amount,UserName,seats_cancelled)values(?,?,?,
?,?,?);

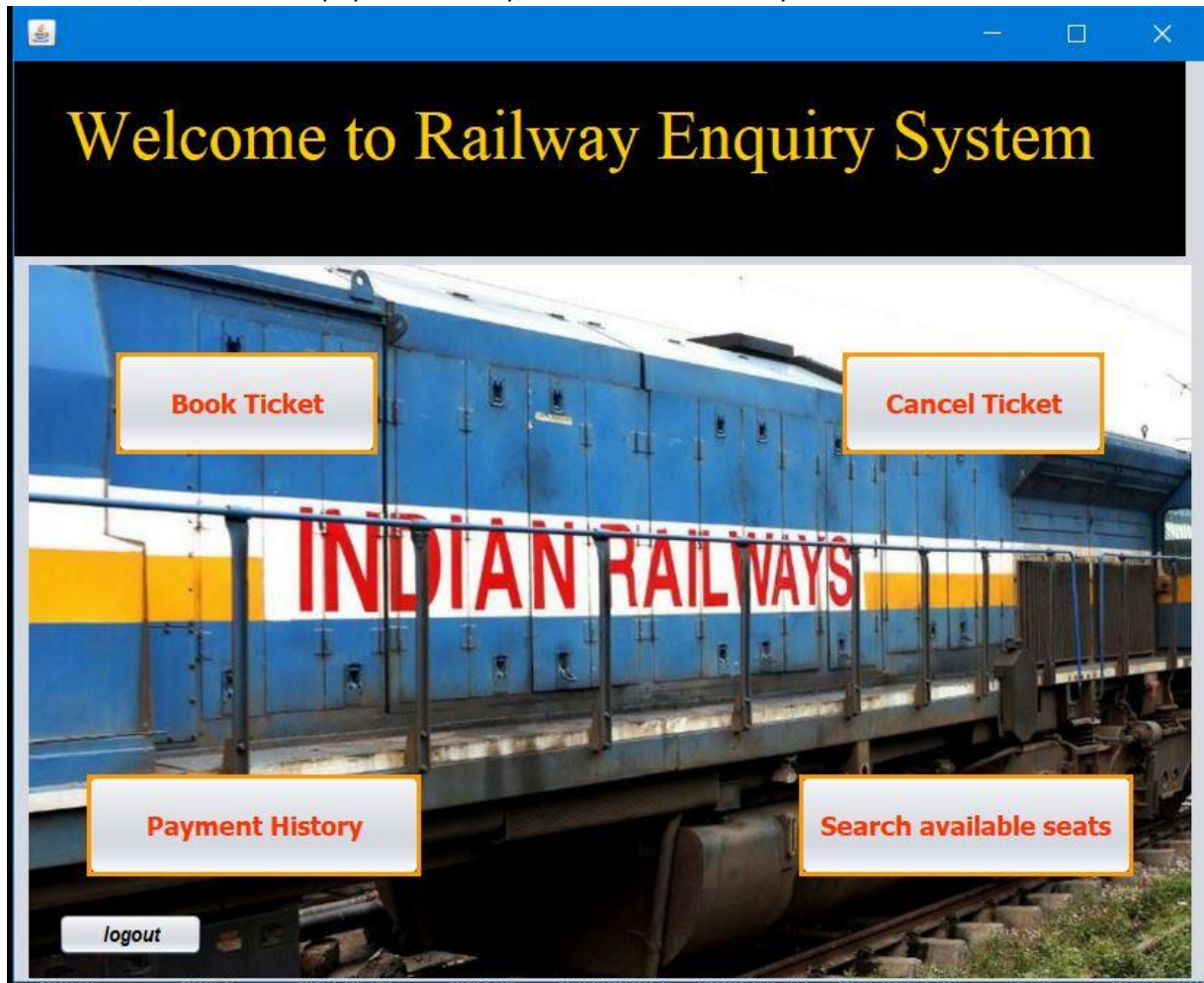
9)Delete from paymentdetails where TicketID='"+ticket.getText()+"';

LoginId=nikhil Pass=1234 then click login



Welcome page with all features:

Book ticket, Cancel Ticket, payment History, Check Seat-Availablity



BookTickets from delhi to Agra with distance 1 so the amount will be 500 for every single ticket and you have to set date on which date you want to book ticket this will not confirm book ticket it is showing available trains and it will give you button to confirm and book ticket on next page

—□×

Book your Tickets here:

From:

To:

Date:

14 ▾

21
22
23
24
25
26
27

4 ▾

2019 ▾

Show Availability of seats

Back

—□×

Book your Tickets here:

From:

To:

Date:

16 ▾

4 ▾

2019 ▾

Show Availability of seats

Back

Select number of seats and book seat and it is also give you message if you have insufficient balance

Available Trains:

Train_No	Train_Name	Arrive_time	Depart_time	seats_Avail	Total_Seats
67911	Delhi_Kerela	7:50 AM	8:15 AM	30	30

Number of seats:

select

Total Fare:-

Wallet Balance:

back

Book

Available Trains:

Train_No	Train_Name	Arrive_time	Depart_time	seats_Avail	Total_Seats
67911	Delhi_Kerela	7:50 AM	8:15 AM	30	30

Number of seats:

2

select

1

2

3

4

Total Fare:-

Wallet Balance:

back

Book

Available Trains:

Train_No	Train_Name	Arrive_time	Depart_time	seats_Avail	Total_Seats
67911	Delhi_Kerela	7:50 AM	8:15 AM	30	30

Message*×*

i

Updated wallet Successfully...

OK

back

Book

Total Fare:-1000

Wallet Balance:16500

Available Trains:

Train_No	Train_Name	Arrive_time	Depart_time	seats_Avail	Total_Seats
67911	Delhi_Kerela	7:50 AM	8:15 AM	30	30

Message*×*

i

UPdated seat Successfully...

OK

back

Book

Total Fare:-1000

Wallet Balance:16500

Available Trains:

Train_No	Train_Name	Arrive_time	Depart_time	seats_Avail	Total_Seats
67911	Delhi_Kerela	7:50 AM	8:15 AM	30	30

Message

i

Ticket Booked and payment deducted Successfully.. see the payment History for details

OK

back

Book

Wallet Balance: 16500

1000


Your last 5 payments will appear here:-

Your Last 5 payment will appear here:-

PayID	TicketID	Train_No	Date	Time	Amount	Balance
2226	18	67911	16/4/2019	12:26:29.894	500	16500

back

Cancel the booked ticket with ticket id =18



The screenshot shows a web application window with a light gray background. At the top, there is a title bar with a small icon on the left and standard window controls (minimize, maximize, close) on the right. The main content area has the heading "Cancellation of a Ticket" in bold black text. Below the heading, the text "Enter you Details here:" is displayed. Underneath, there is a label "Enter you Ticket ID:" followed by a text input field containing the value "18". A button labeled "Confirm Cancel" is positioned below the input field. In the bottom-left corner of the window, there is a button labeled "back".

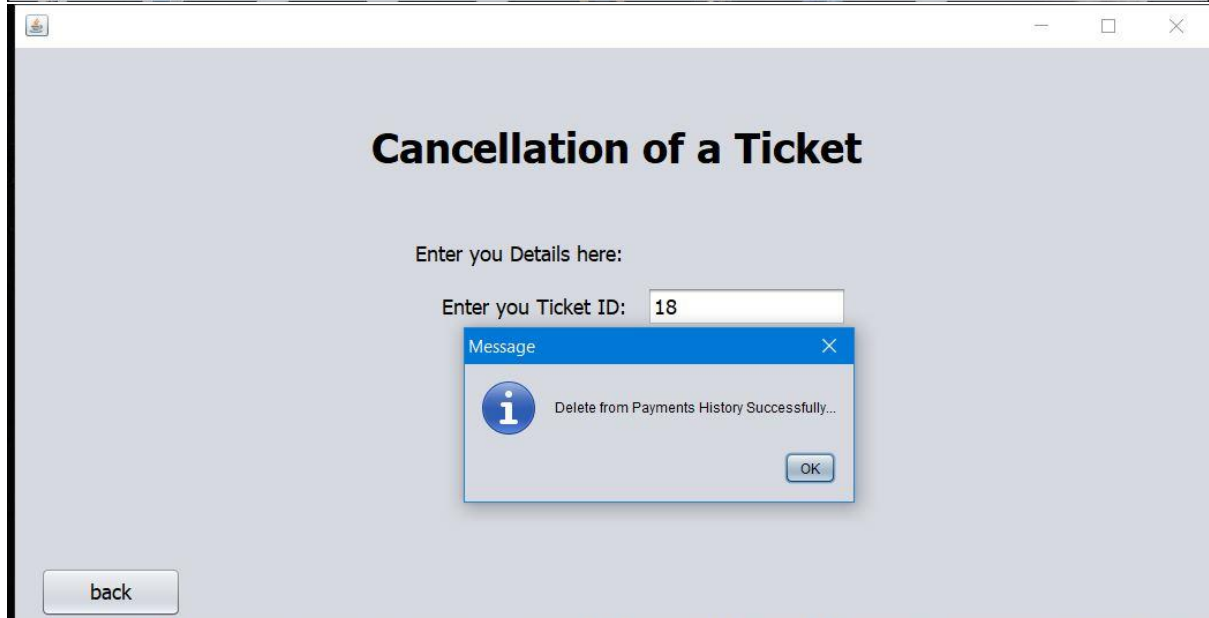
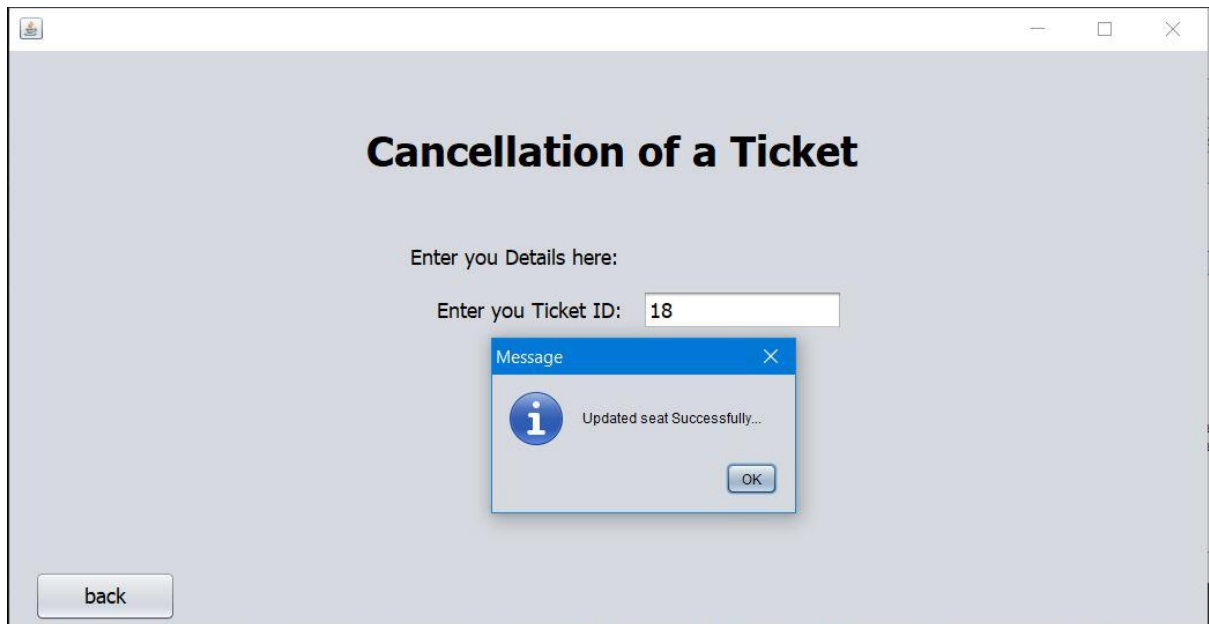
Cancellation of a Ticket

Enter you Details here:

Enter you Ticket ID:

Confirm Cancel

back



Payment history for that particular ticket also deleted correctly

Your Last 5 payment will appear here:-

PayID	TicketID	Train_No	Date	Time	Amount	Balance
-------	----------	----------	------	------	--------	---------

[back](#)

Cancelled records updated successfully in database

Showing rows 0 - 1 (2 total, Query took 0.0010 seconds.)

`SELECT * FROM `cancelled_records``

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	PayID	TicketID	Train_No	Amount	UserName	seats_cancelled
<input type="checkbox"/> Edit Copy Delete	2226	18	67911	500	nikhil	2
<input type="checkbox"/> Edit Copy Delete	7192	20	88056	500	nikhil	1

☐ Check all | With selected: Edit Copy Delete Export

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

You can check availability of seats before booking the tickets

Check seat availability

Enter Train no.

[Check](#)

[back](#)

Register new user if already exists then show message

The image shows two screenshots of a web application's user registration interface. The top screenshot shows the registration form with the following fields: "Name" (filled with "nikhil"), "Create Password" (filled with "****"), and "Confirm Password" (filled with "****"). A "Register" button is located below the fields. At the bottom, there is a link "Fill Permanent Details" and a note: "If you want to book any ticket you must fill Details here:". The bottom screenshot shows the same form, but with an error message displayed in a modal dialog box. The dialog box has a title "Message" and a blue information icon. The message text is "UserName already Exist please try another...". There is an "OK" button at the bottom right of the dialog box.

Register as a new User

Name:


Create Password:

Confirm Password:

[Register](#)

If you want to book any ticket you must fill Details here: [Fill Permanent Details](#)

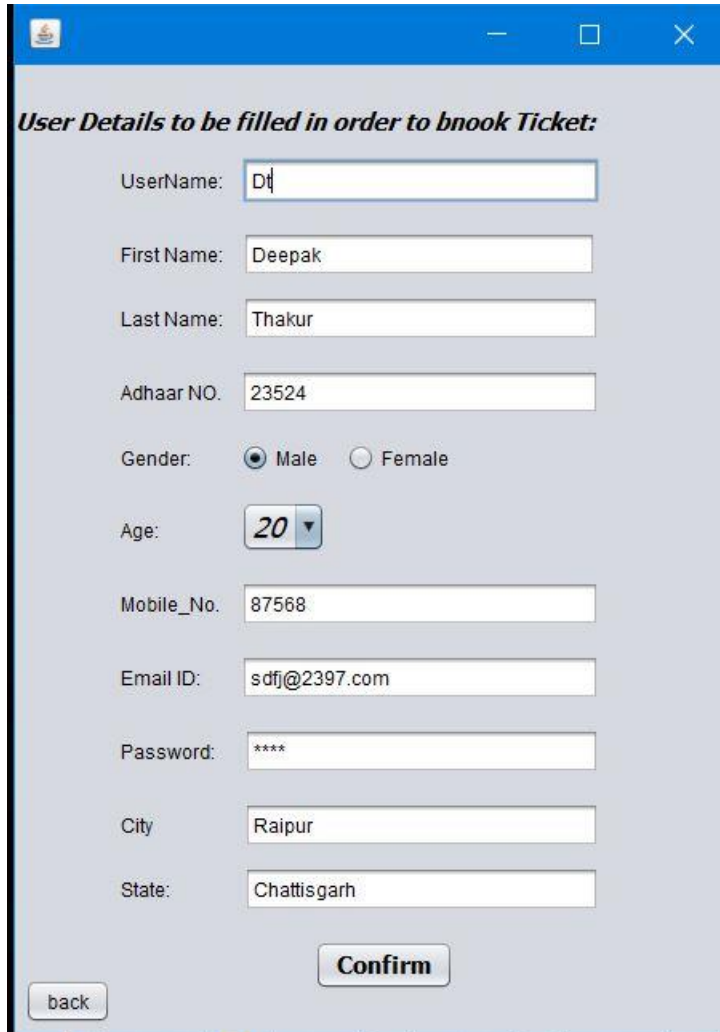
Message

 UserName already Exist please try another...

[OK](#)

If you want to book any ticket you must fill Details here: [Fill Permanent Details](#)

Registering new user details for booking tickets



The image shows a web browser window with a registration form. The form is titled "User Details to be filled in order to bnook Ticket:". It contains several input fields for user information, including Username, First Name, Last Name, Adhaar NO., Gender (radio buttons for Male and Female), Age (a dropdown menu), Mobile_No., Email ID, Password, City, and State. At the bottom, there are "back" and "Confirm" buttons.

User Details to be filled in order to bnook Ticket:

UserName:

First Name:

Last Name:

Adhaar NO.

Gender: ☒ Male ☐ Female

Age:

Mobile_No.

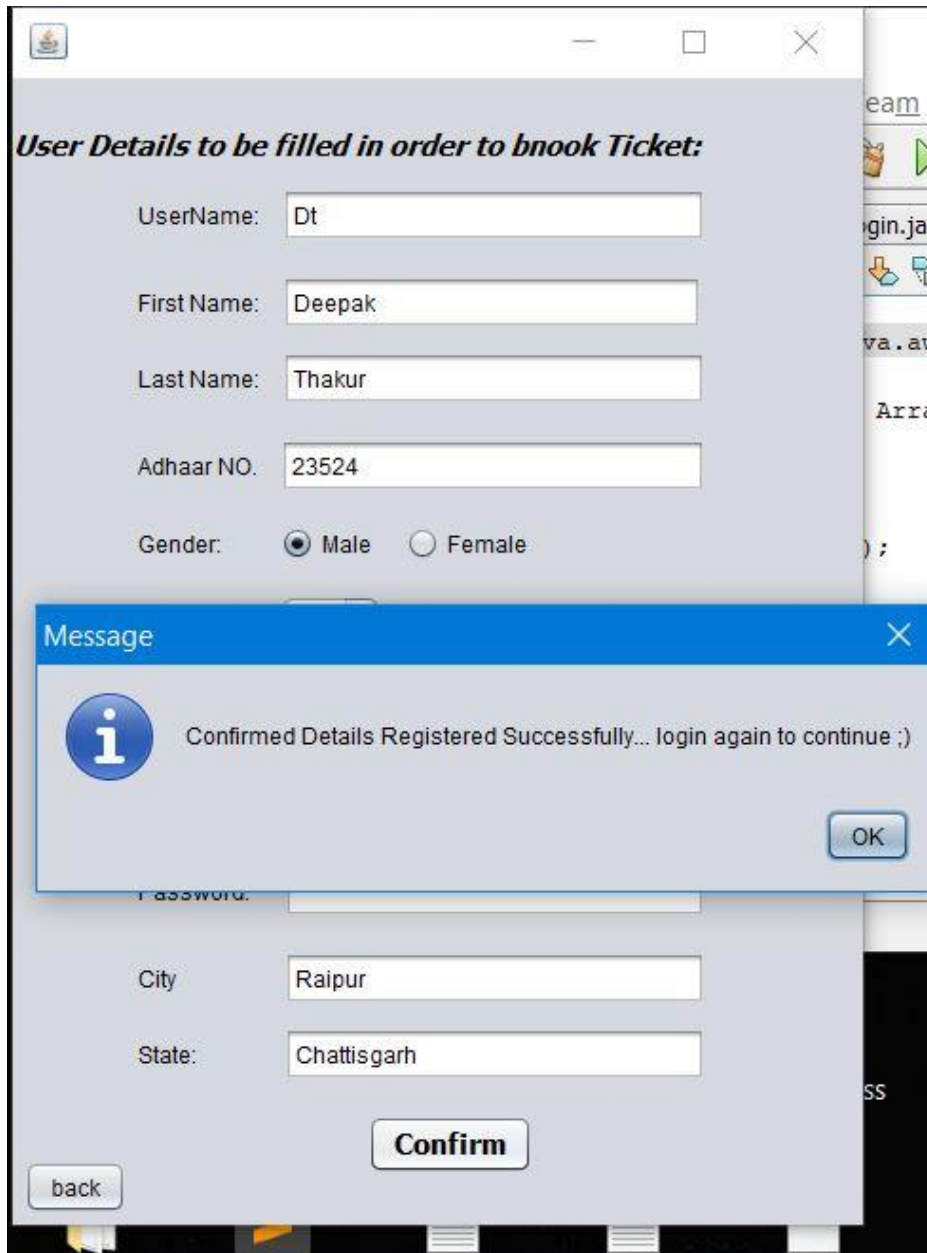
Email ID:

Password:

City

State:

Confirmation message



The image shows a Java Swing window titled "User Details to be filled in order to book Ticket:". The window contains several text input fields and a radio button group. A modal message dialog is displayed in the foreground, indicating successful registration.

User Details to be filled in order to book Ticket:

UserName:

First Name:

Last Name:


Adhaar NO:

Gender: ☒ Male ☐ Female

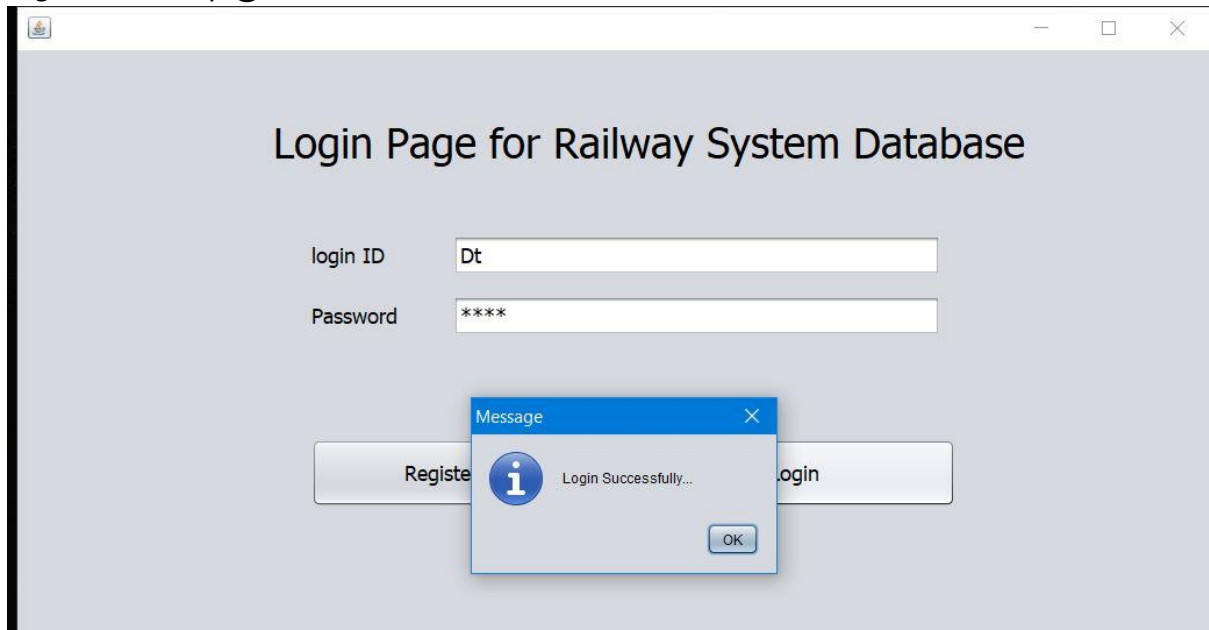
City:

State:

Message

 Confirmed Details Registered Successfully... login again to continue ;)

Login Successfully 😊



The screenshot shows a web browser window titled "Login Page for Railway System Database". The page has a light blue background. At the top, the title "Login Page for Railway System Database" is displayed in a large, bold, black font. Below the title, there are two input fields: "login ID" with the value "Dt" and "Password" with the value "****". Below these fields are two buttons: "Register" on the left and "Login" on the right. A modal message box is centered over the buttons. The message box has a blue header with the text "Message" and a close button (X). The main content of the message box shows an information icon (i) followed by the text "Login Successfully...". At the bottom of the message box is an "OK" button.

Login Page for Railway System Database

login ID Dt

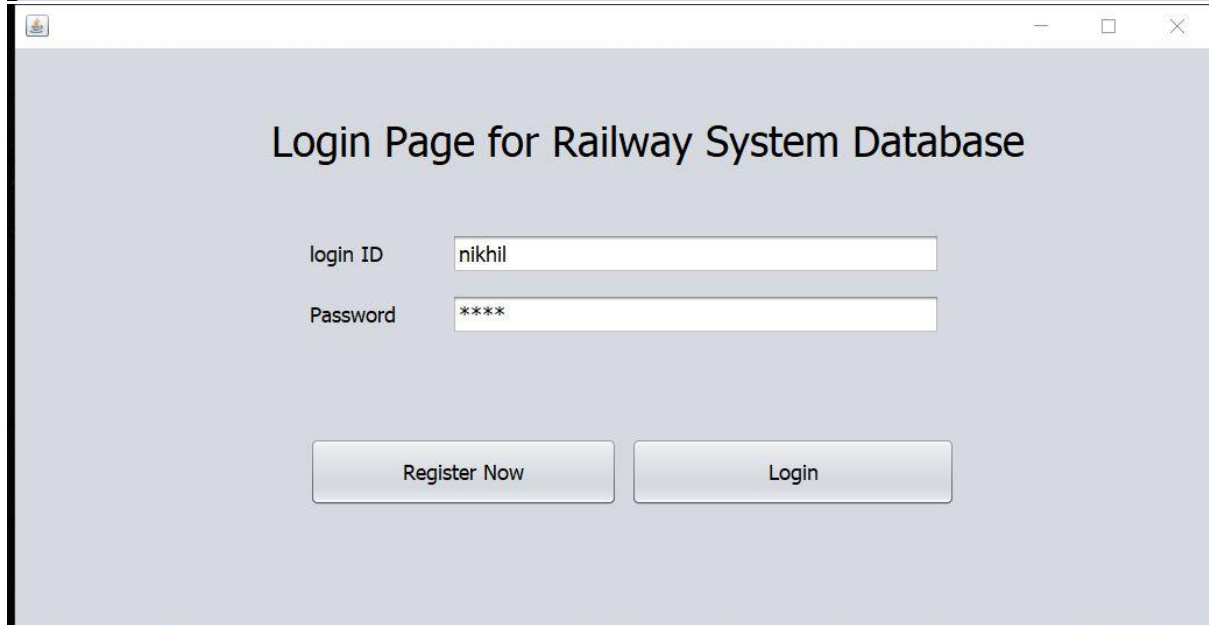
Password ****

Register Login

Message

Login Successfully...

OK



The screenshot shows the same web browser window as the previous one, but with the modal message box removed. The "login ID" field now contains the value "nikhil" and the "Password" field still contains "****". The "Register Now" and "Login" buttons are still present at the bottom of the form area.

Login Page for Railway System Database

login ID nikhil

Password ****

Register Now Login