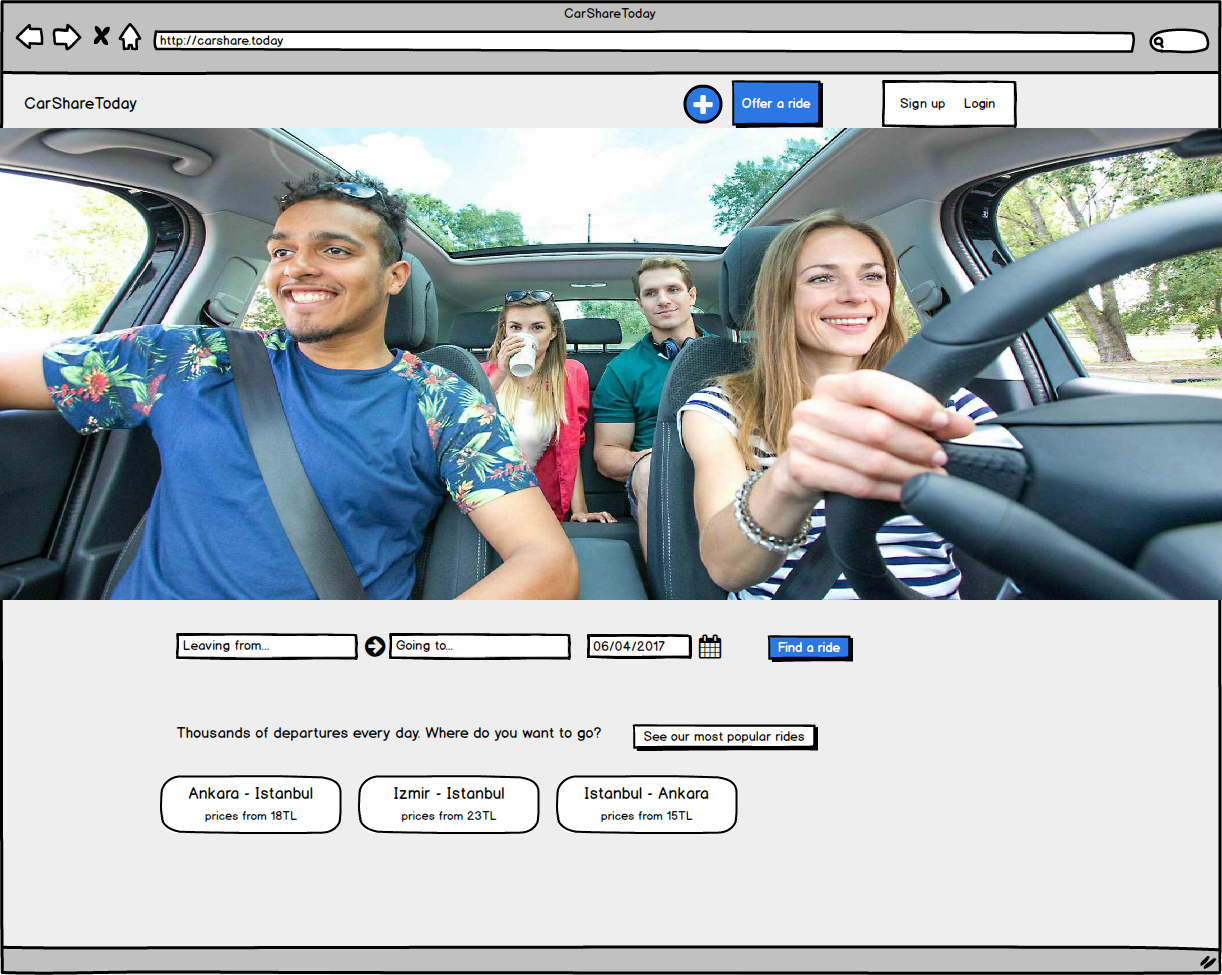
Home



Process: The home screen is used by both driver and the passenger. When the screen is loaded, 5 main features which consist of ride offering, sign up, login, ride finding and seeing most popular rides.

Inputs: @leaving, @going, @date

**SQL**

**Find a ride**

select pickup\_location, destination, date, driver\_email

from ScheduledRide

where pickup\_location = ‘@leaving’ and destination = ‘going’ and date = ‘date’

**See most popular rides**

select pickup\_location, destination, date, driver\_email, count(pickup\_location) as occurrence

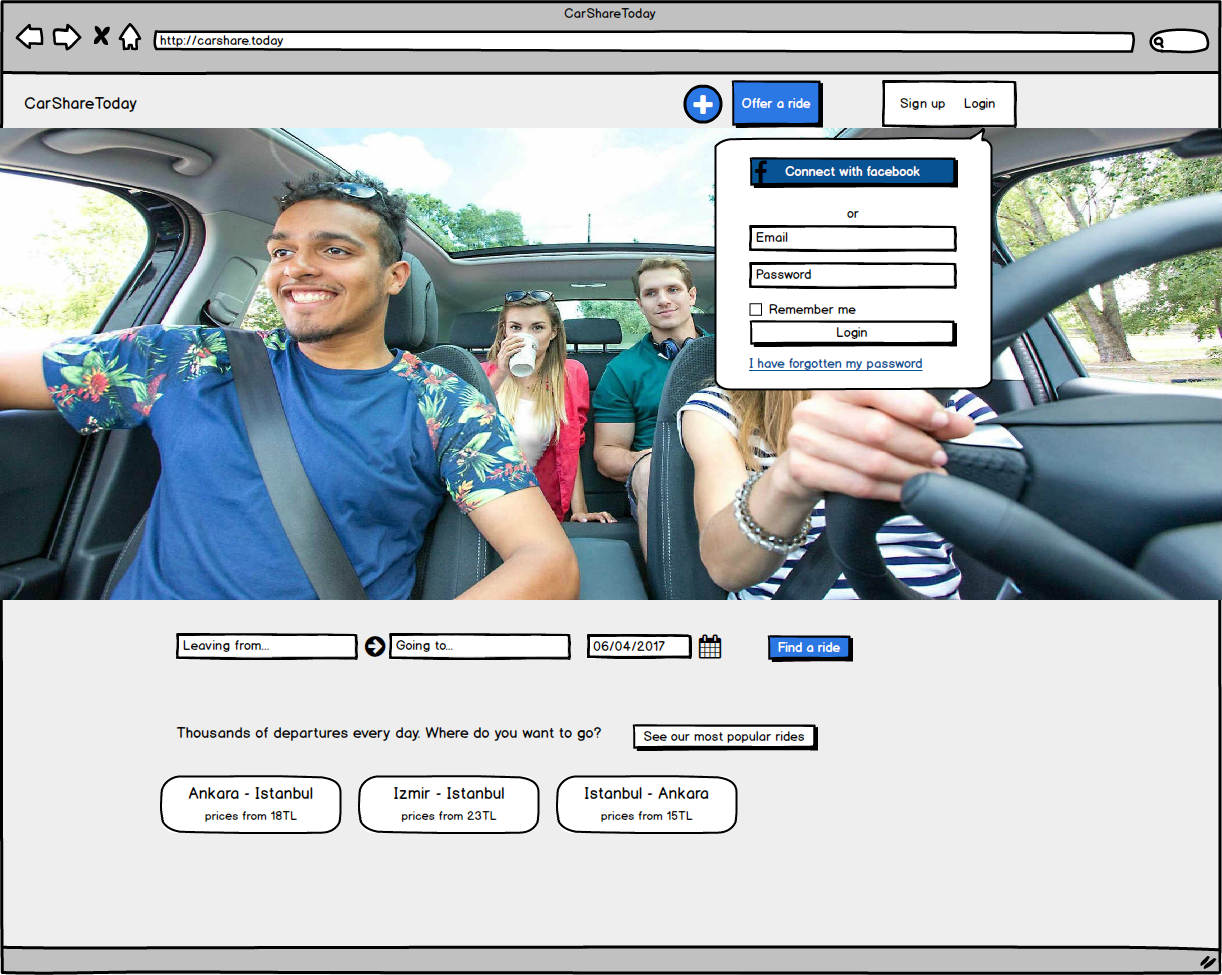
from ScheduleRide

group by pickup\_location

order by occurrence DESC

limit 20;

Login



Process: Besides the processes of main menu, in this screen the user can login his/her existing account.

Inputs: @email, @password

SQL:

Login:

If exists (select id

from User

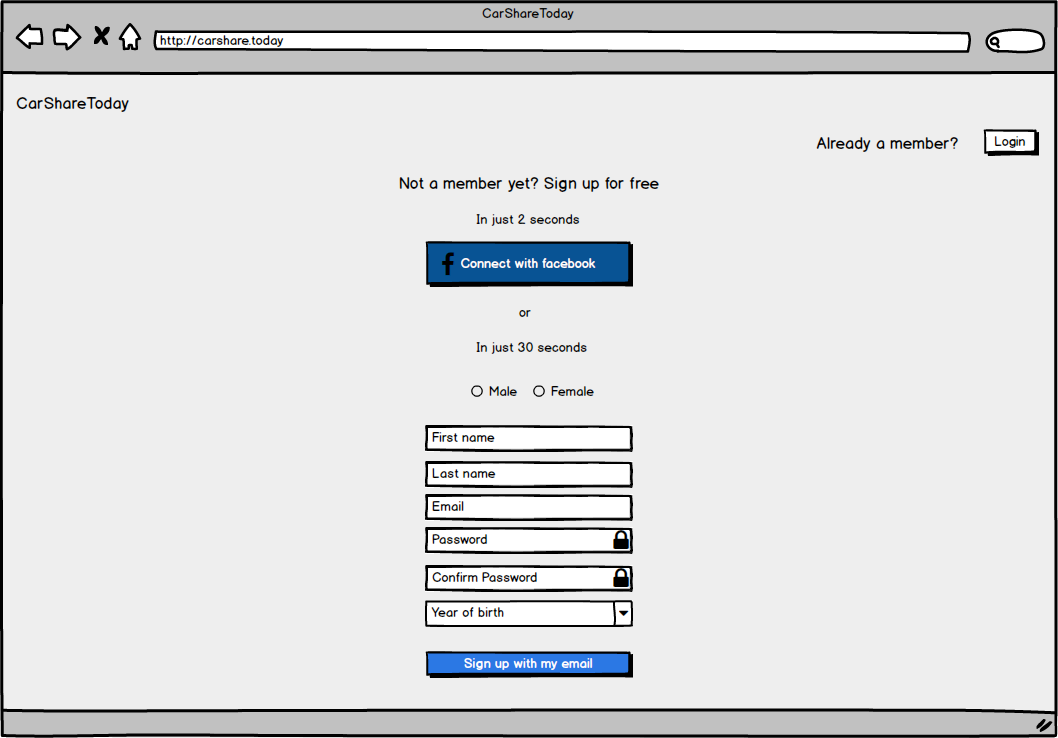
where email = ‘@email’ and password = ‘@password’ )

begin

// login successful

End

Sign Up



Process: The sign up screen provides user to enroll to the system with some specific properties of the user.

Inputs: @gender, @fname, @lname, @email, @password, @confirm, @age

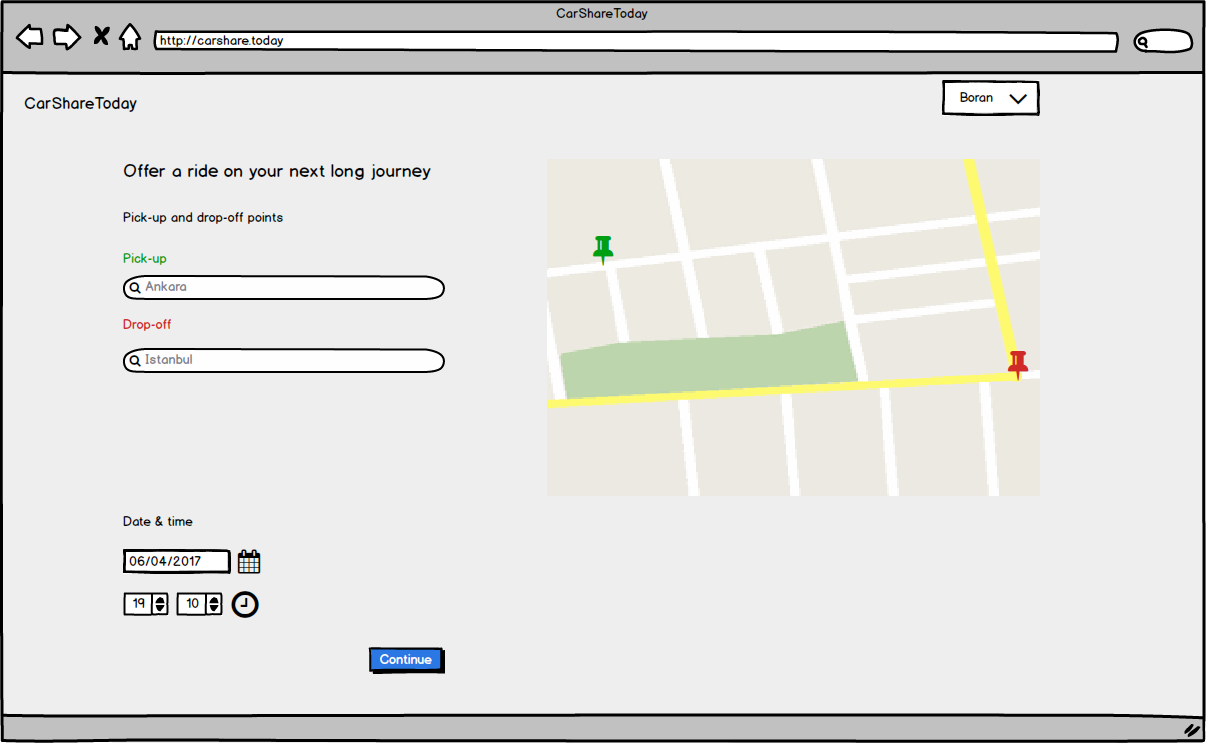
SQL:

Signup

Insert into User (id, gender, fname, lname, email, password, confirm, age)

Values(0123456789, @gender, @fname, @lname, @email, @password, @age)

Offer a ride

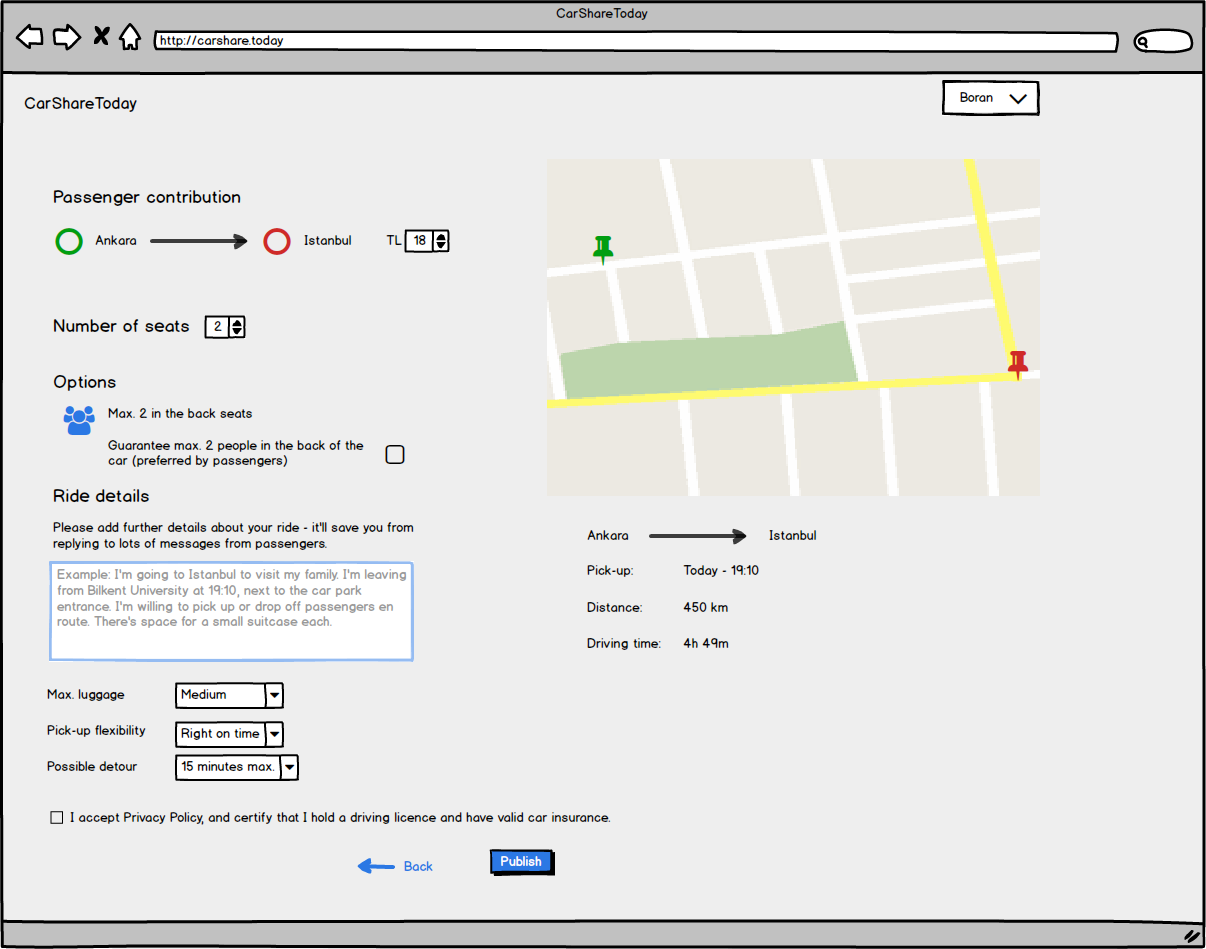


Process: The rider can generate a new ride from this screen by adding pick up and drop off locations and date and time. Google Maps shows that the route and locations of the offered ride.

Inputs: @pickup, @destination, @date

SQL Part is generated in the next page.

Offer a ride details



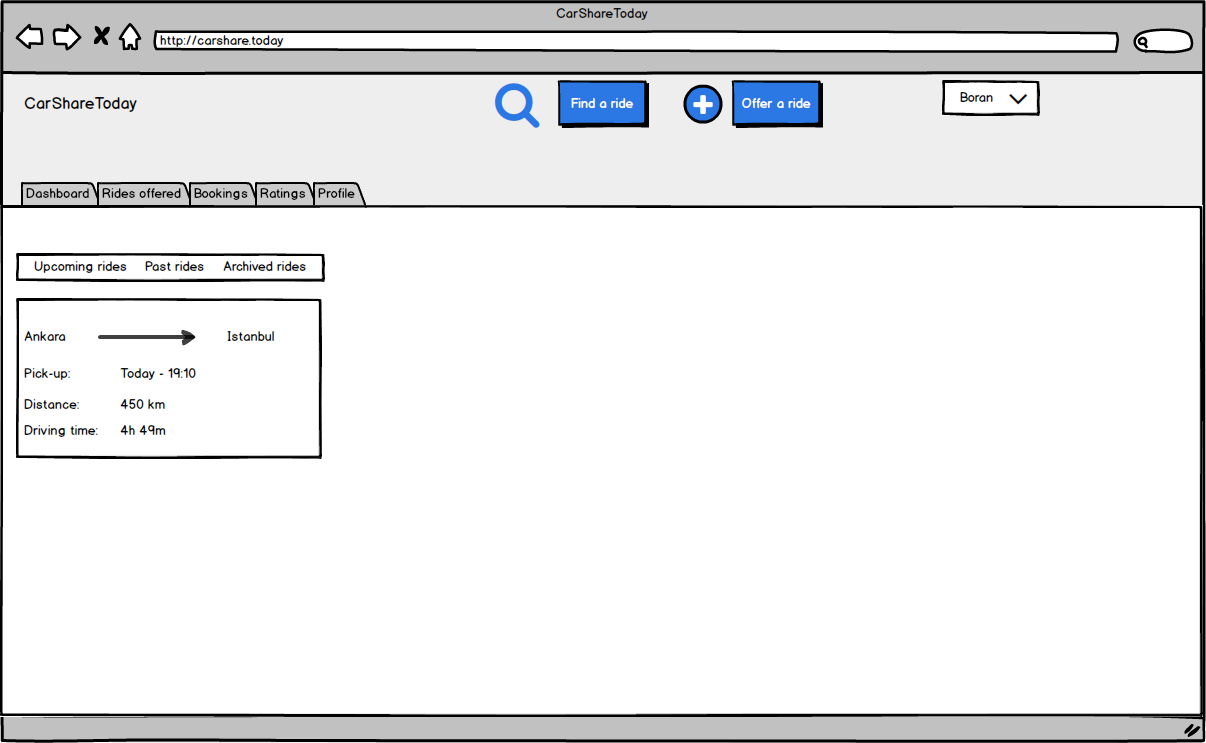
Process: After the choosing of the locations and time are chosen this screen appears. Some details about the ride are specified such as number of seats, luggage, time flexibility and price.

Inputs: @seats, @details, @luggage, @pickup\_flexibility, @detour

Insert into(ride\_id, number\_of\_empty\_seats, date, pickup\_location, destination, driver\_email, details, max\_luggage, pickup\_flexibility, possible\_detaur)

Values(123456789, @seats, @date, @pickup, @destination, current\_user\_email, , @details, @luggage, @pickup\_flexibility, @detour)

Upcoming rides



Process: User can see the upcoming rides which are offered by him/her.

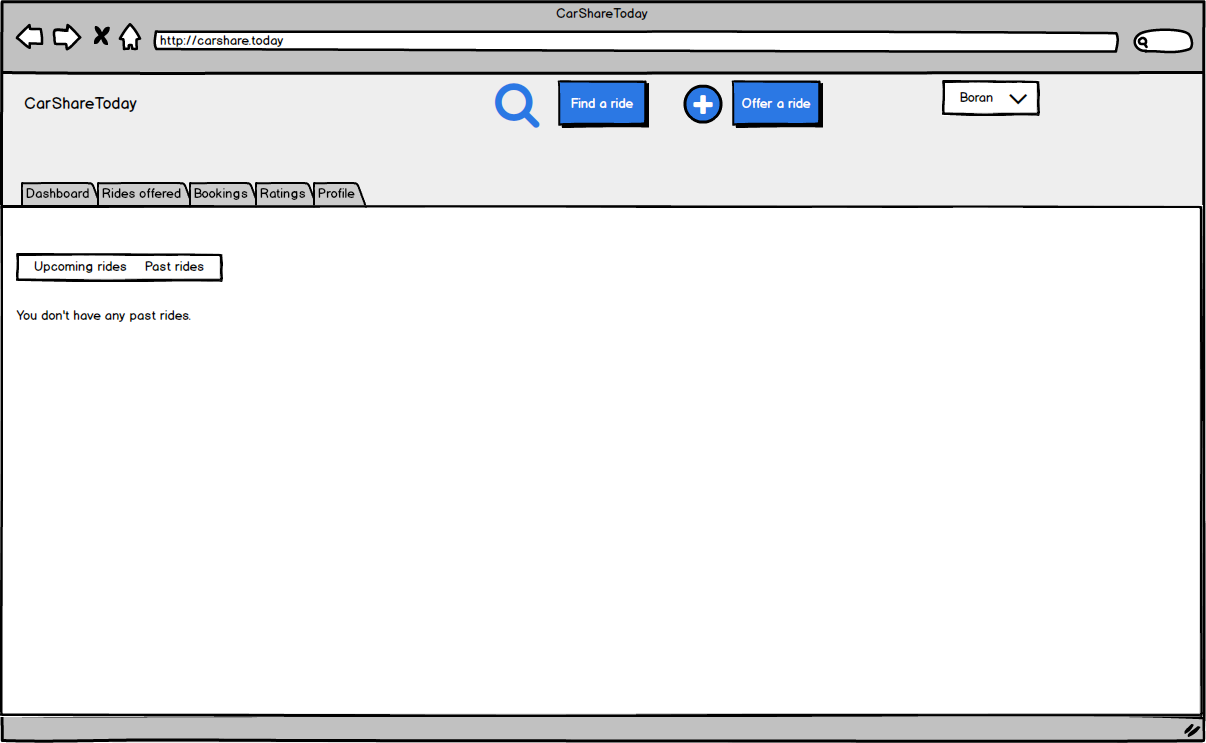
SQL:

Select \*

From ScheduledRide

Where driver\_email = ‘current\_user\_email’

Past offered rides



Process: User can see the past rides which are offered by him/her.

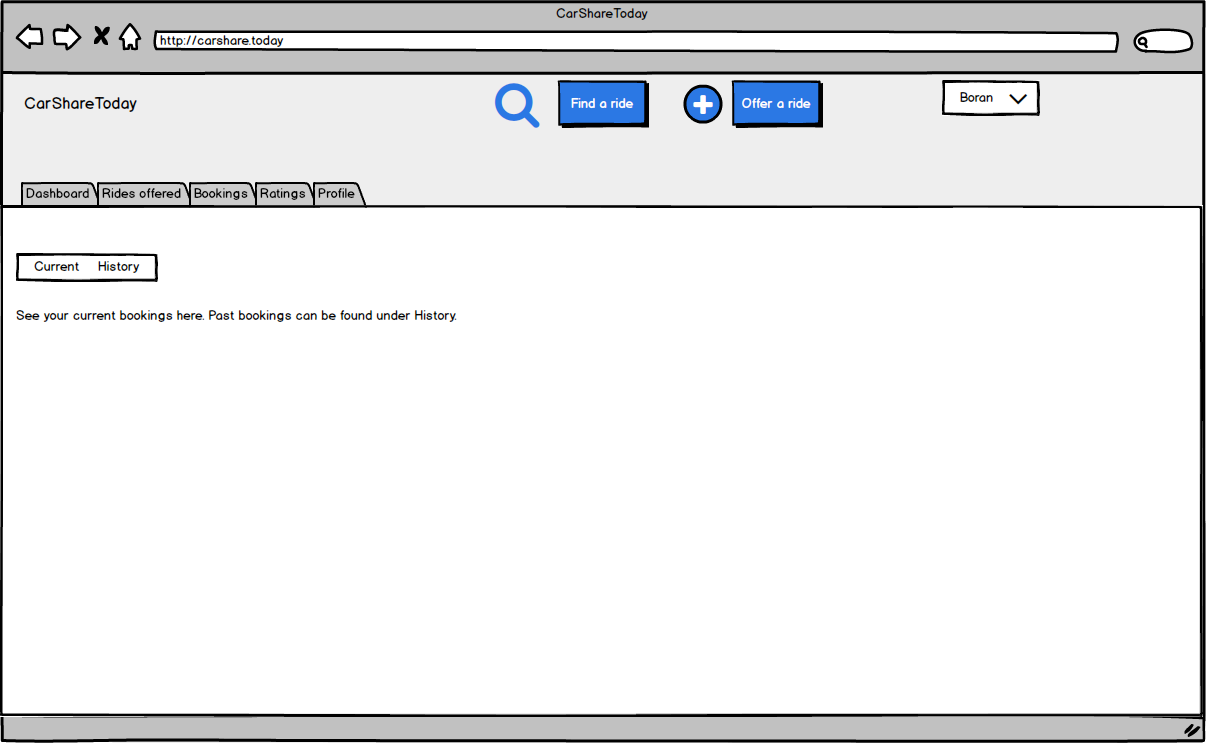
SQL:

Select \*

From CompleteRide

Where driver\_email = ‘current\_user\_email’

Current Bookings



Process: User can see the current rides which are booked by him/her.

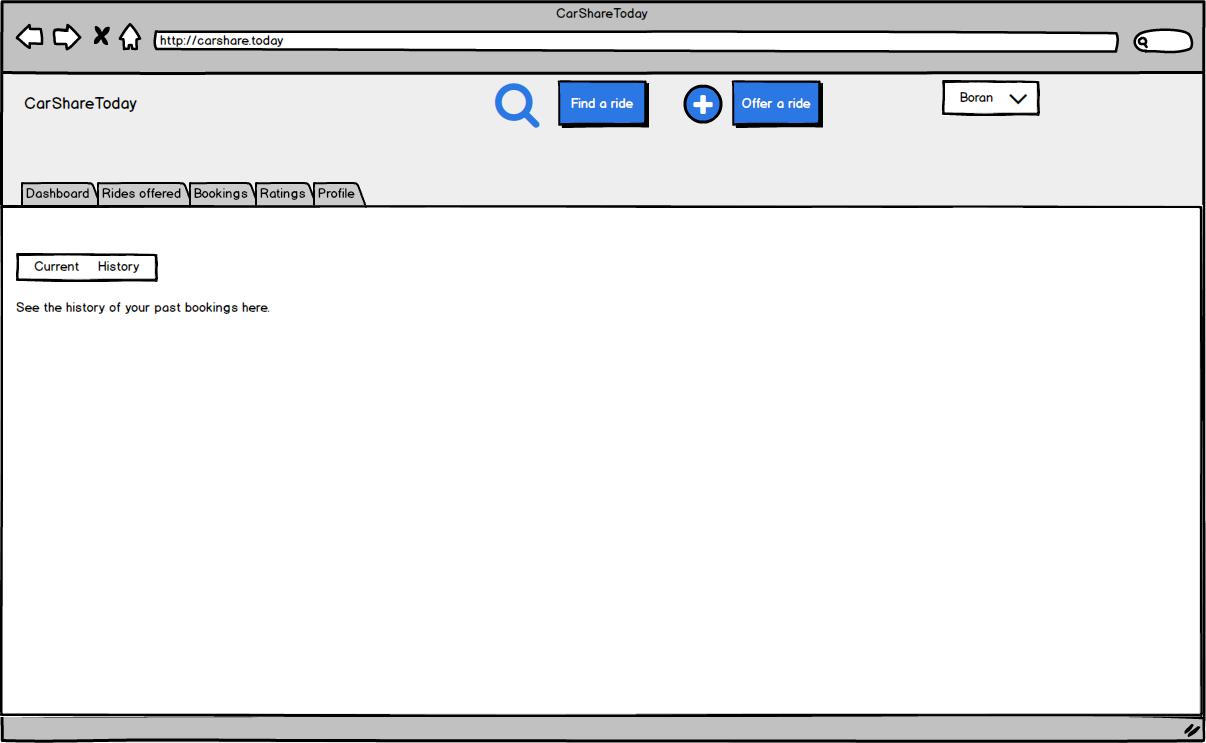
SQL:

Select ride\_id

From creates natural join Request natural join Reserves

Where userID = ‘current\_user\_id’

Past bookings



Process: User can see the past rides which were booked by him/her.

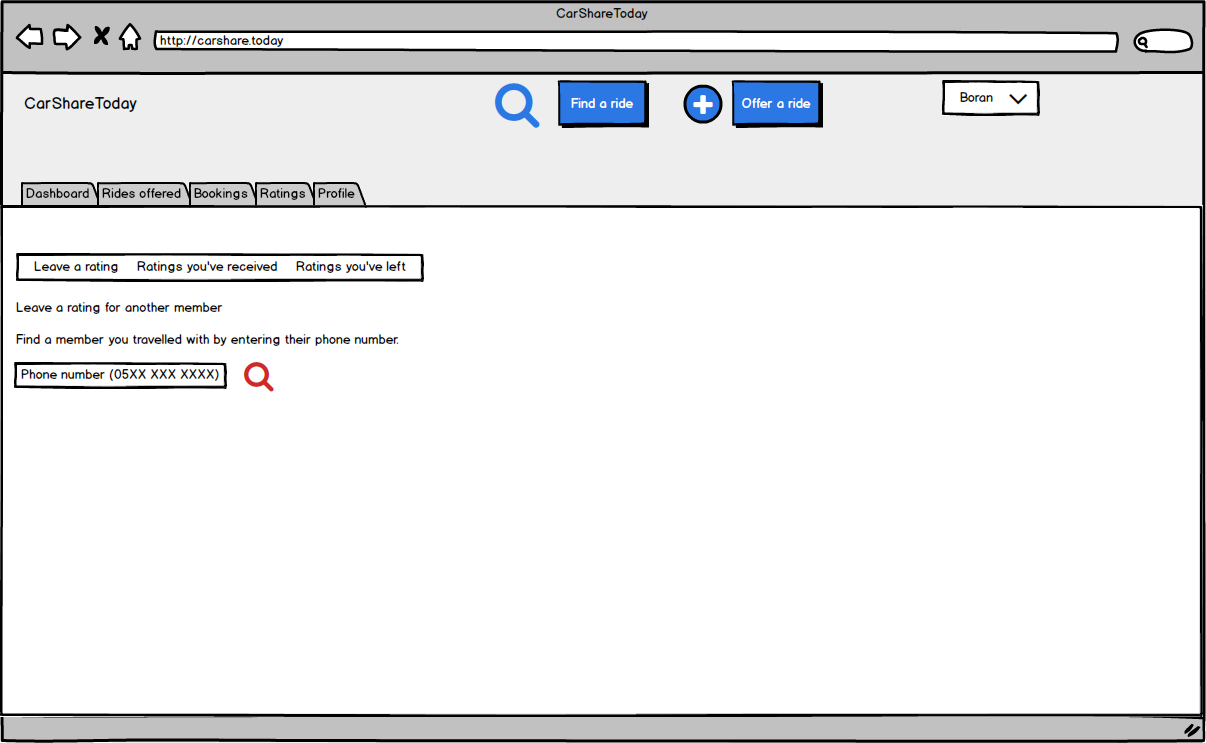
SQL:

Select ride\_id

From creates natural join Request natural join CompleteRide

Where userID = ‘current\_user\_id’

Leave a rating



Process: User can leave a rating for a person by entering the phone number of the user who will be rated.

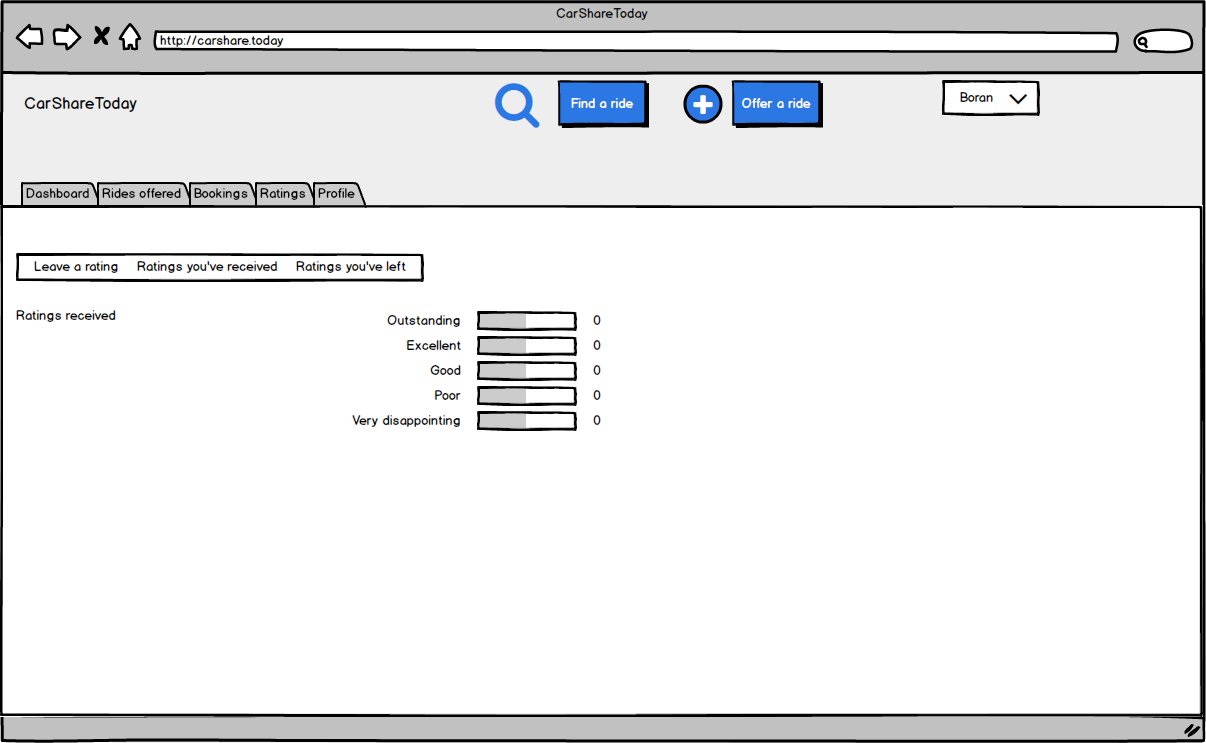
Input: @phone

Select userID

From User

Where phone\_num = ‘@phone’

Received Ratings



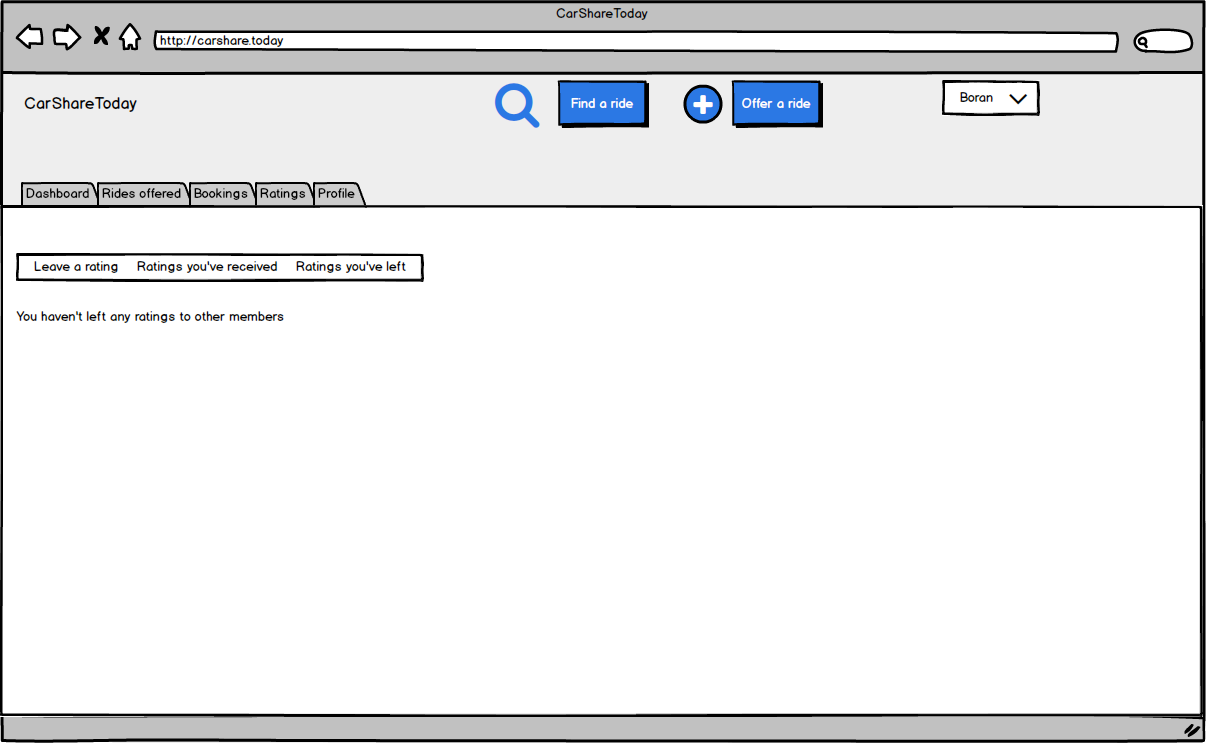
Process: User can see the ratings about him/her.

Select rating

From Ratings natural join Account\_Ratings

Where reviewed\_id = ‘current\_user\_id’

Ratings left by user



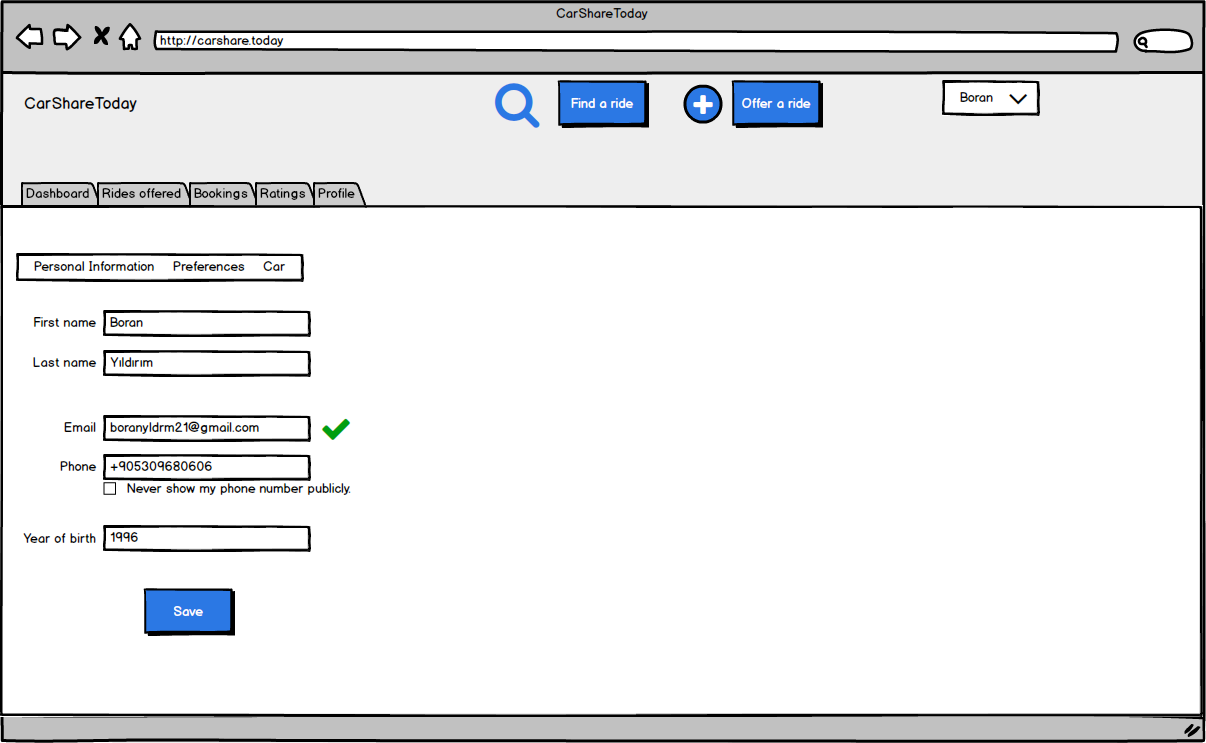
Process: User can see the ratings which are written by him/her.

Select rating

From Ratings natural join Account\_Ratings

Where reviewer\_id = ‘current\_user\_id’

Personal Information



Process: This screen provides opportunity to change the profile information of the user.

Inputs: @fname, @lname, @email, @phone, @age

SQL:

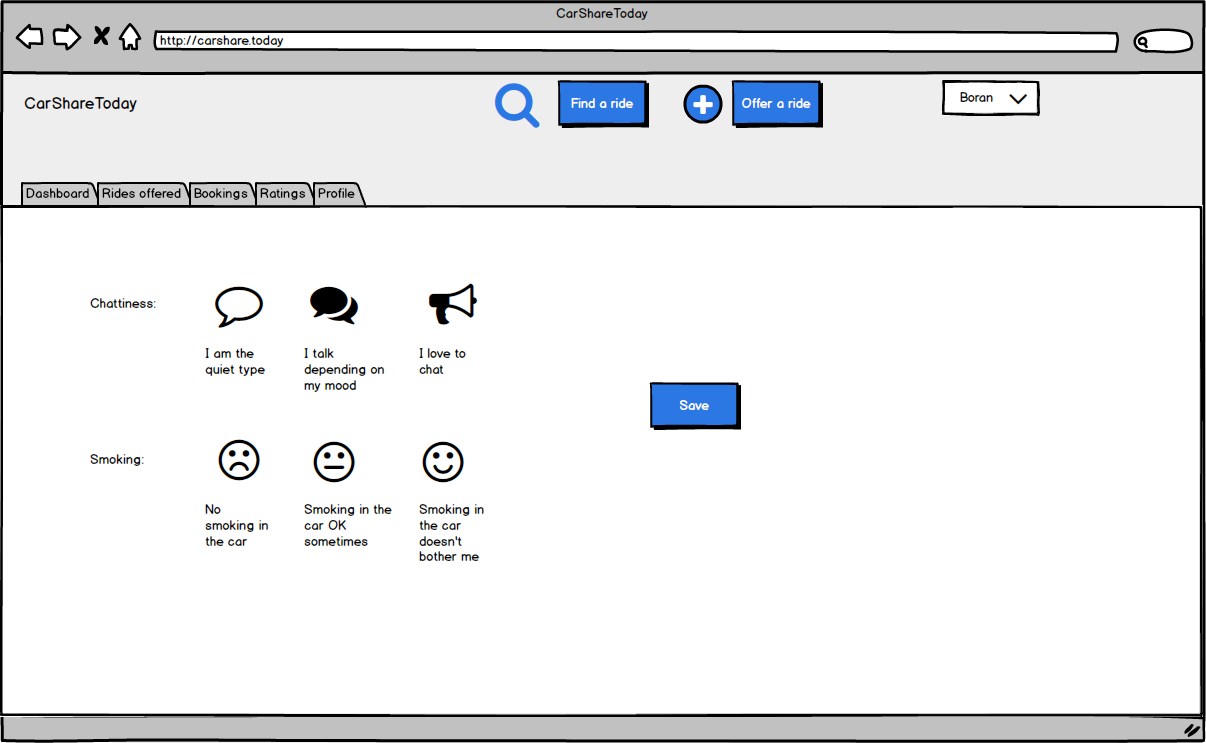
Save:

Update User

Set fname = ‘@fname’, lname = ‘@lname’, email = ‘@email’, phone\_num = ‘@phone’, age = ‘@age’

Where userID = ‘current\_user\_ID’

Ride Preferences



Process: User can change the preferences about ride and use them as a filter when searching a ride.

Inputs: @chattiness, @smoking

SQL:

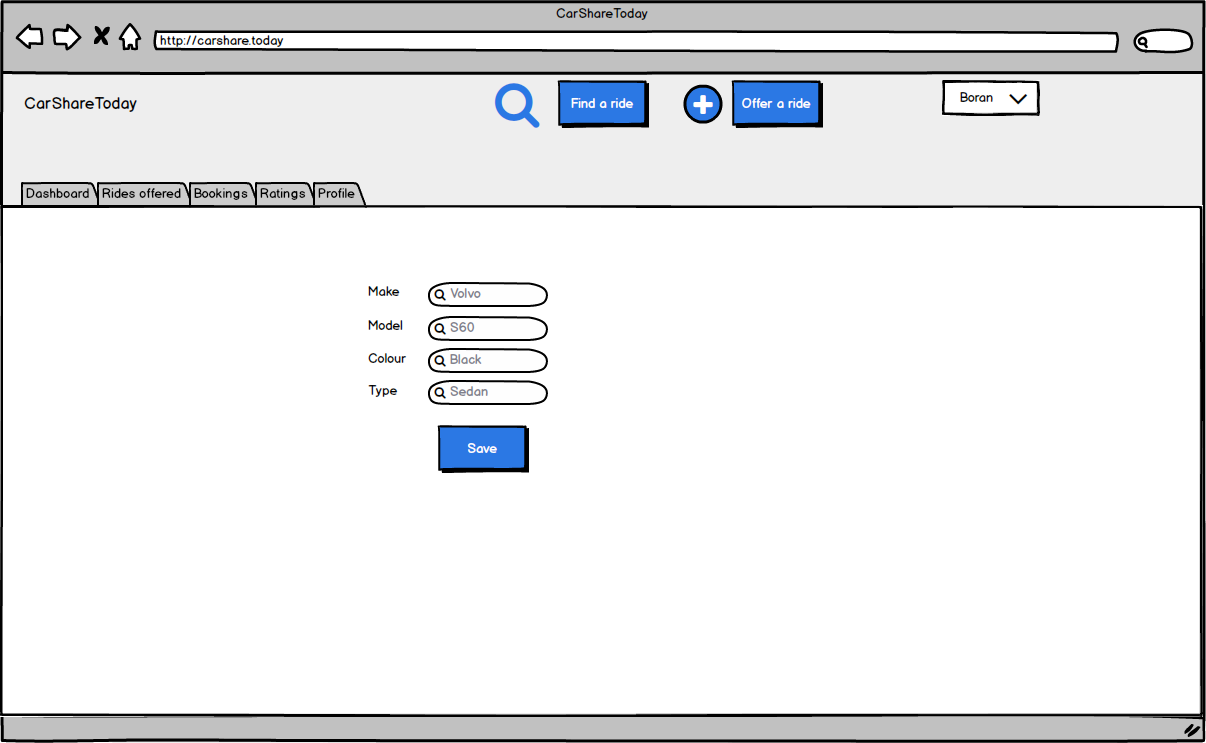
Save:

Update User

Set chattiness = ‘@chattiness’, smoking = ‘@smoking’

Where userID = ‘current\_user\_ID’

Car



Process: From this screen user can add a car to the system or change the car which s/he owns.

Inputs: @make, @model, @colour, @type

SQL:

Save:

If exists(select \* from Car natural join Owns where userID = ‘current\_user\_ID)

Begin

Update Car

Set brand = ‘@make’, model = ‘@model’, colour = ‘@colour’, type = ‘@type’

End

Else

Begin

Insert into Car(brand, model, colour, type)

Values(@make, @model, @colour, @type)

End