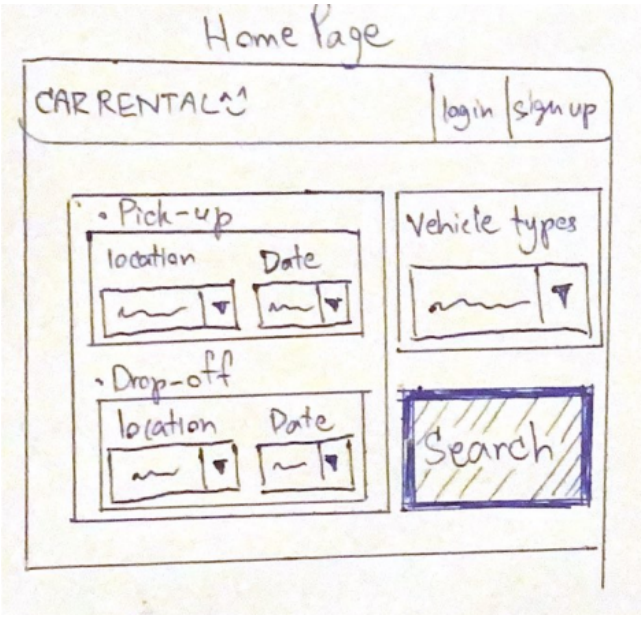
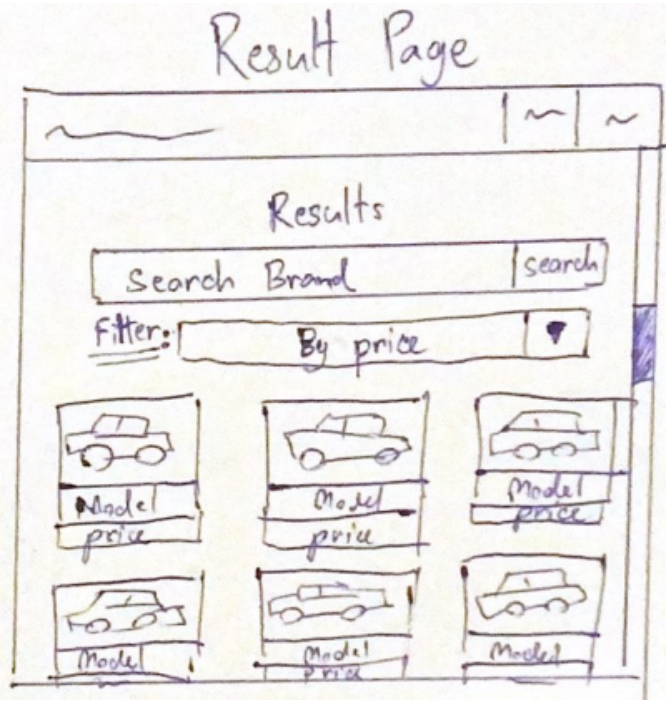


Start with the homepage where all users have to visit as the landing page, each user who is interested in a car needs to specify pick-up location, drop-off location, and vehicle type.

Then after he;/she presses the search button. They will see the result page.



The result is shown by a data query based on the user's inputs.
For example, if user selected **current_location of 1001 Henderson St, Fort Worth, TX** and **the type of `economy`**

```
%%sql
SELECT v.*
FROM vehicle v
JOIN location l ON v.current_location_id = l.id
JOIN vehicle_type vt ON v.vehicle_type_id = vt.id
WHERE l.street_address = '1001 Henderson St'
  AND l.city = 'Fort Worth'
  AND l.state = 'TX'
  AND vt.name = 'Economy';
```

```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
1 rows affected.
```

id	brand	model	model_year	mileage	color	vehicle_type_id	current_location_id	rental_price
1	Nissan	Versa	2016	65956	white	1	1	1200.50

In reality there are many car rental companies that, regardless of how distant you are, will deliver the car to your pick-up for free. If that is the case, then **the results will only depend on vehicle type** as below.

```
%%sql
SELECT v.*
FROM vehicle v
JOIN vehicle_type vt ON v.vehicle_type_id = vt.id
WHERE vt.name = 'Economy';
```

```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
2 rows affected.
```

id	brand	model	model_year	mileage	color	vehicle_type_id	current_location_id	rental_price
1	Nissan	Versa	2016	65956	white	1	1	1200.50
2	Mitsubishi	Mirage	2017	55864	light blue	1	6	1350.00

Now, we will talk about the Result Page. The result page shows choices of cars based on your pre-defined criteria from the Home Page. Below the car image, it must show the **brand, model, and rental_price.**

In case you select **all types**, rather than picking one, these are the results you will get.

```
%%sql

SELECT brand, model, rental_price
FROM vehicle;
```

```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
6 rows affected.
```

brand	model	rental_price
Nissan	Versa	1200.50
Mitsubishi	Mirage	1350.00
Chevrolet	Cruze	2059.99
Hyundai	Elantra	2999.89
Volkswagen	Jetta	4000.00
Toyota	RAV4	2555.25

Furthermore, if you have a look right above all car images, you will see that **users can also search by preferred brand.**

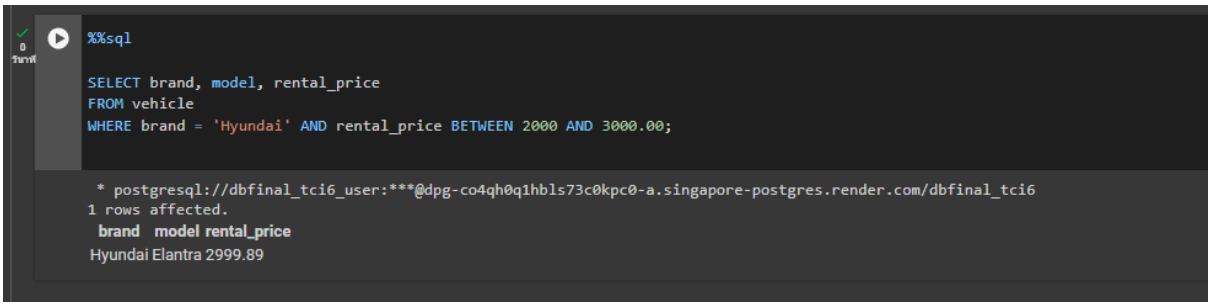
For example, if they pick **Hyundai.**

```
%%sql

SELECT brand, model, rental_price
FROM vehicle
WHERE brand = 'Hyundai';
```

```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
1 rows affected.
 brand  model rental_price
Hyundai Elantra 2999.89
```

Users can also do layers of filtration. On top of brand, they may search by price (usually the choice comes as ranges of prices). Let's say I want a **Hyundai at the rental price range between 2000 to 3000.**

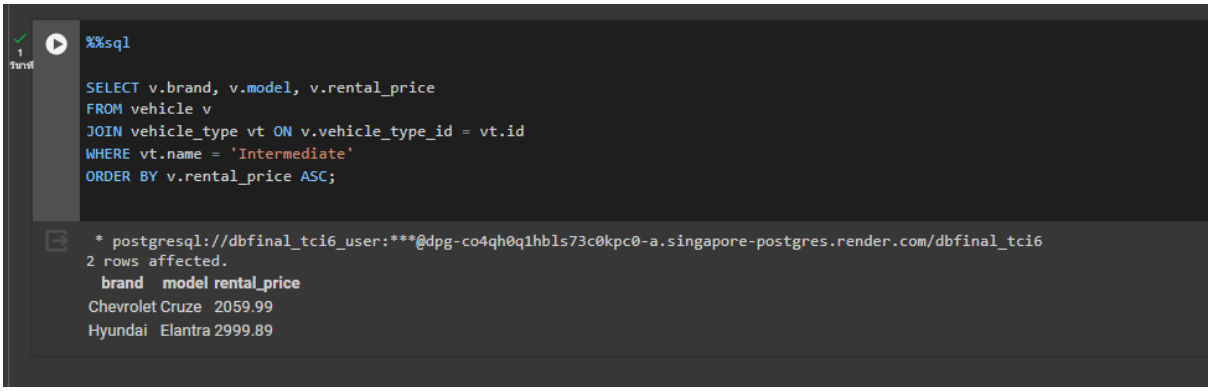


```
%%sql

SELECT brand, model, rental_price
FROM vehicle
WHERE brand = 'Hyundai' AND rental_price BETWEEN 2000 AND 3000.00;
```

```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
1 rows affected.
 brand  model rental_price
Hyundai Elantra 2999.89
```

I am not in to eco-cars, so I want an **Intermediate vehicle** type with rental **price ascendingly sorted**.

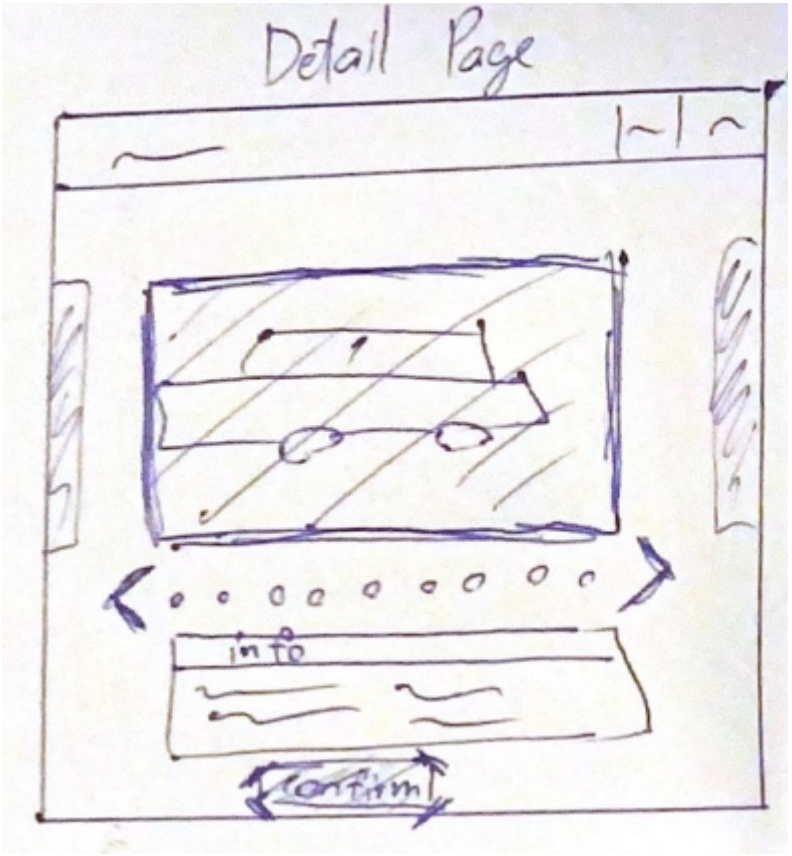


```
%%sql

SELECT v.brand, v.model, v.rental_price
FROM vehicle v
JOIN vehicle_type vt ON v.vehicle_type_id = vt.id
WHERE vt.name = 'Intermediate'
ORDER BY v.rental_price ASC;
```

```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
2 rows affected.
  brand  model rental_price
Chevrolet Cruze  2059.99
Hyundai Elantra 2999.89
```

Let's say that the user has found the right car. They will click on the image of the selected car. Then it comes to the next page.



This is the Detail Page of the selected vehicle. A lot of images from many dimensions illustrate the car condition. Below the image, it is a full list of car's information.

Since the car is already picked, we can query by vehicle's id to display all the information.

```
%%sql

SELECT
    v.brand,
    v.model,
    v.model_year,
    v.mileage,
    v.color,
    v.rental_price,
    vt.name AS vehicle_type_name,
    CONCAT(l.street_address, ', ', l.city, ', ', l.state, ' ', l.zipcode) AS location_address
FROM
    vehicle v
JOIN
    vehicle_type vt ON v.vehicle_type_id = vt.id
JOIN
    location l ON v.current_location_id = l.id
WHERE
    v.id = 2;
```

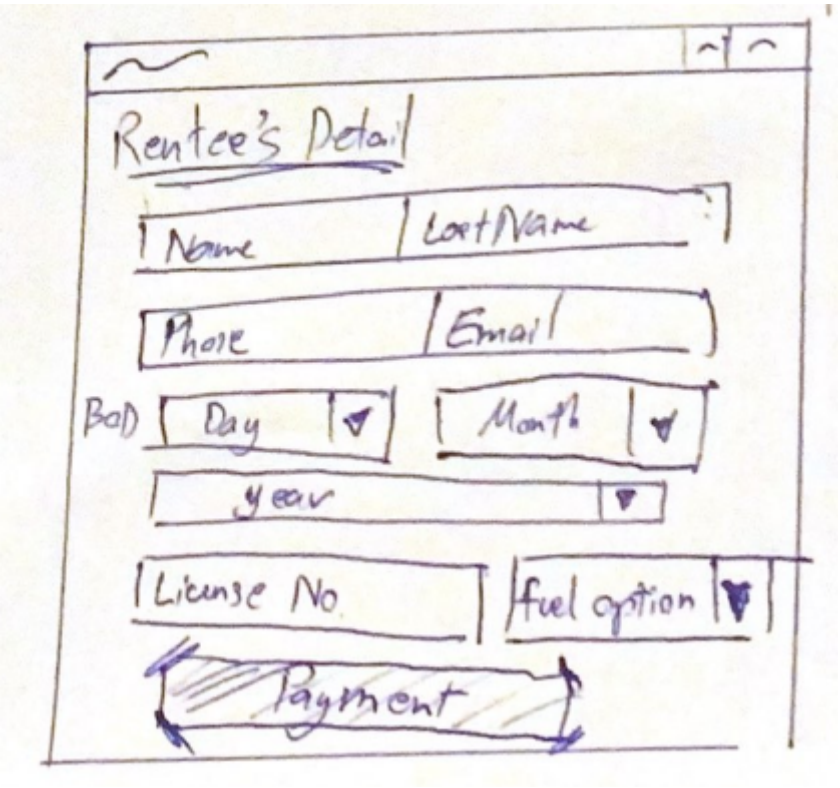
```
 * postgresql://dbfinal_tci6_user:***@dpg-co4qh0q1hbls73c0kpc0-a.singapore-postgres.render.com/dbfinal_tci6
1 rows affected.
```

brand	model	model_year	mileage	color	rental_price	vehicle_type_name	location_address
Mitsubishi	Mirage	2017	55864	light blue	1350.00	Economy	1011 Pike St, Seattle, WA 42345

Note: in this code, we do not display any `id` because users have no ideas what they all mean and make the information looks more confusing, hence bad for user experience, so we extract only human-readable info.

Next. If you are satisfied with the result you see, click Confirm.

Then, you will move to the Credential Page, where you have to fill your personal information.



You will specify all the info on this page, and have to select the fuel option of your rent.

In reality there are choices whether you will pay for the gas or not. There are three choices, **prepay**: pay up front, **self-service**: pay for your own gas and have to left the gas at the same level when the car is out, **Market**: you will buy gas at the market price, which is not determined by the car rental company and not a part of your rental package.

One restriction to go further is you have to meet the **legal driving age.** Otherwise, you cannot rent a car.

```
-- to filter out underage driver
CREATE OR REPLACE FUNCTION age_check_function()
RETURNS TRIGGER AS $$
BEGIN
  IF (NEW.birth_date > (CURRENT_DATE - INTERVAL '21 years')) THEN
    RAISE EXCEPTION 'You are too young to drive bro~';
  END IF;
  RETURN NEW;
END;
$$ LANGUAGE plpgsql;

-- trigger
DROP TRIGGER IF EXISTS age_check ON customer;
CREATE TRIGGER age_check
BEFORE INSERT ON customer
FOR EACH ROW
EXECUTE FUNCTION age_check_function();
```

What could be improved are that:

1. There is no payment system or invoices returned to the customer.
2. Once complete the renting process and payment, all the rental details should be added to the system and made the car unavailable for a duration, so other renters cannot select this car.
3. There should be a choice of electric cars, not only petrol ones.