

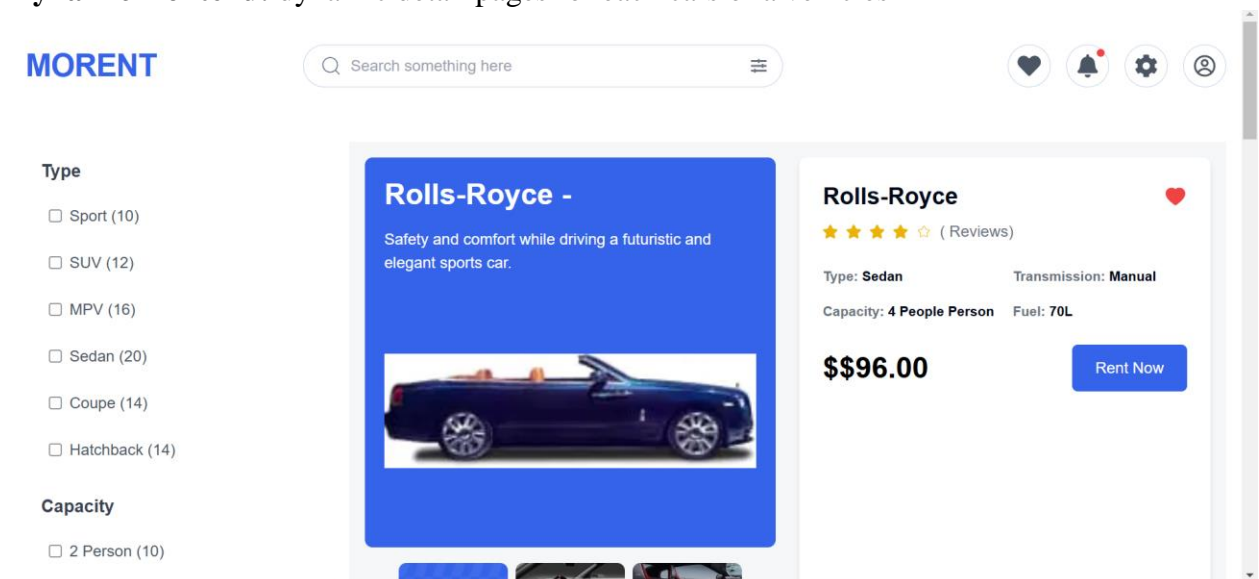
Day 5: Testing, Error Handling, and Backend Integration — [by: AHMED RAZA]

1. Objective

The primary aim of Day 5 was to refine the back-end processes of the car rental platform. This involved improving real-time data management, implementing effective error-handling strategies, and integrating Stripe for seamless payment transactions. Additionally, thorough testing was performed to ensure the platform remains stable and provides a smooth user experience.

2. Testing

- ❖ **Frontend Testing:** The header includes essential element such as the logo, account access and wish list for login user. The navigation menu for proper functionality. The banner and product carousels display and navigate correctly.
- ❖ **Dynamic frontend:** dynamic detail pages for each cars or a vehicles



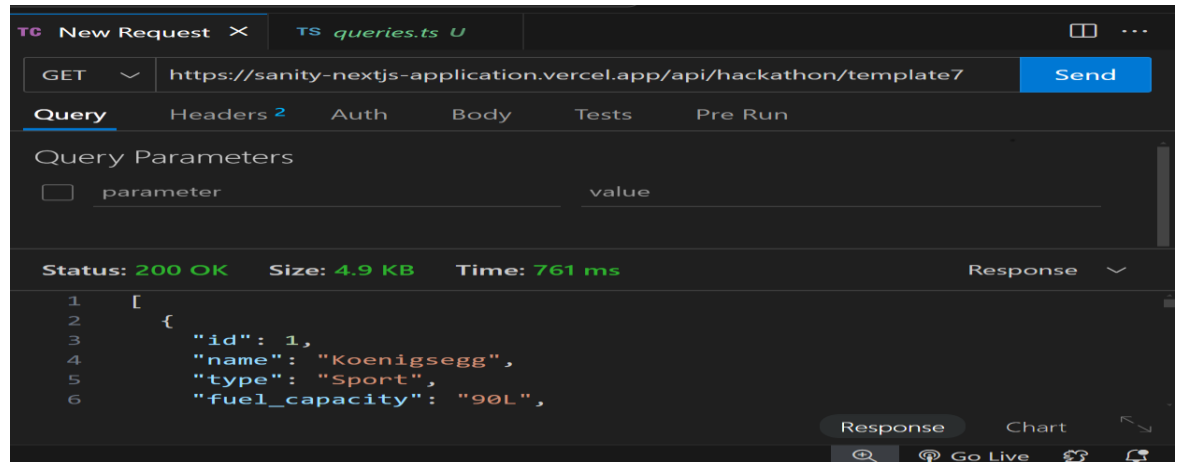
- ❖ **Integration Testing:** Ensured seamless communication between the front end and backend, specifically in handling data and API requests.
- ❖ **Clerk authentication:** sign in through different way like a google, email and Facebook etc.
- ❖ **Payment System Testing:** Evaluated the Stripe integration by testing transaction processes and handling failed payment scenarios.

3. Error Handling Implementation

- **Frontend Error Management:**

- Displays user-friendly messages when API requests fail.
- Ensures form submission of rent and payment only proceed when all required fields are completed.
- My data is perfectly fetch from API & and the callback UI works perfectly

○

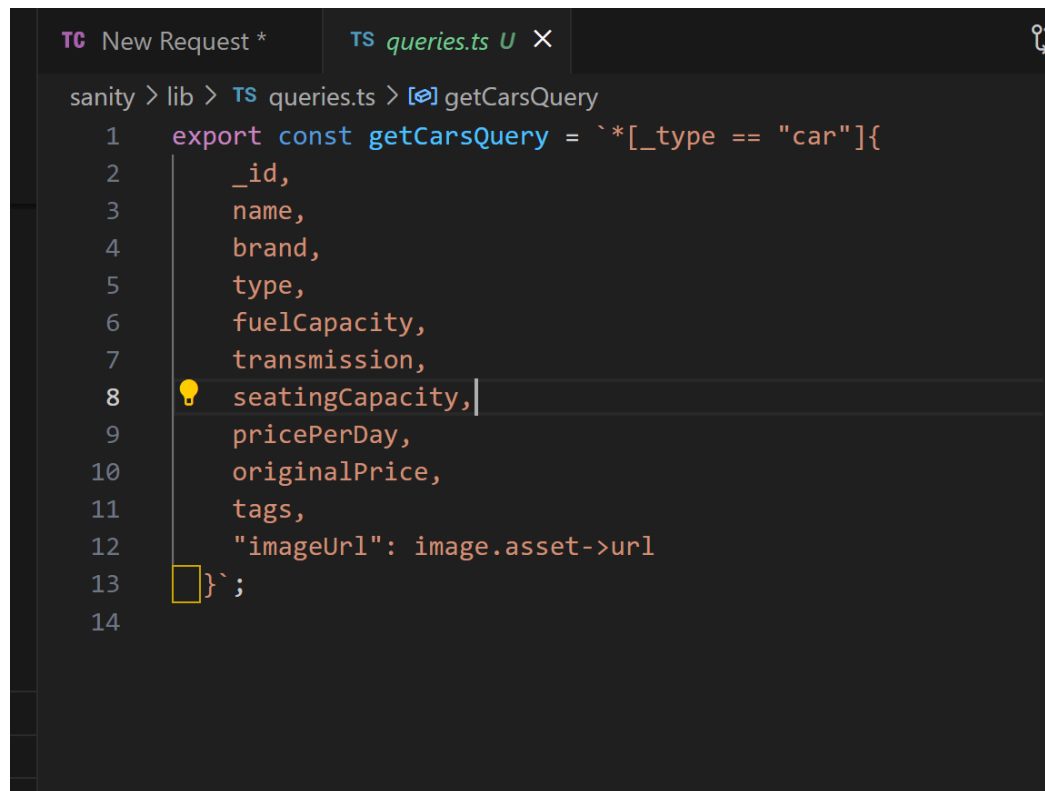


○

4. Back end Integration Refinement

- **Real-Time Data Management:**

- Backend dynamically updates car availability, pricing, and booking slots using Sanity CMS, ensuring the front end always displays the latest data without requiring manual updates.



The screenshot shows a code editor with a dark theme. At the top, there are two tabs: 'New Request *' and 'queries.ts U X'. The main editor area displays a TypeScript file named 'queries.ts' with the following code:

```
sanity > lib > TS queries.ts > [getCarsQuery]
1  export const getCarsQuery = `*[_type == "car"]{
2      _id,
3      name,
4      brand,
5      type,
6      fuelCapacity,
7      transmission,
8      seatingCapacity,
9      pricePerDay,
10     originalPrice,
11     tags,
12     "imageUrl": image.asset->url
13 }`;
14
```

A yellow lightbulb icon is visible next to line 8, indicating a suggestion or warning. The cursor is positioned at the end of line 8, after the comma following 'seatingCapacity'.

- **Stripe Payment Integration:**

- Payments are securely processed through Stripe, verifying transactions before confirming bookings.

Transactions

All12

Succeeded3

+ Date and time

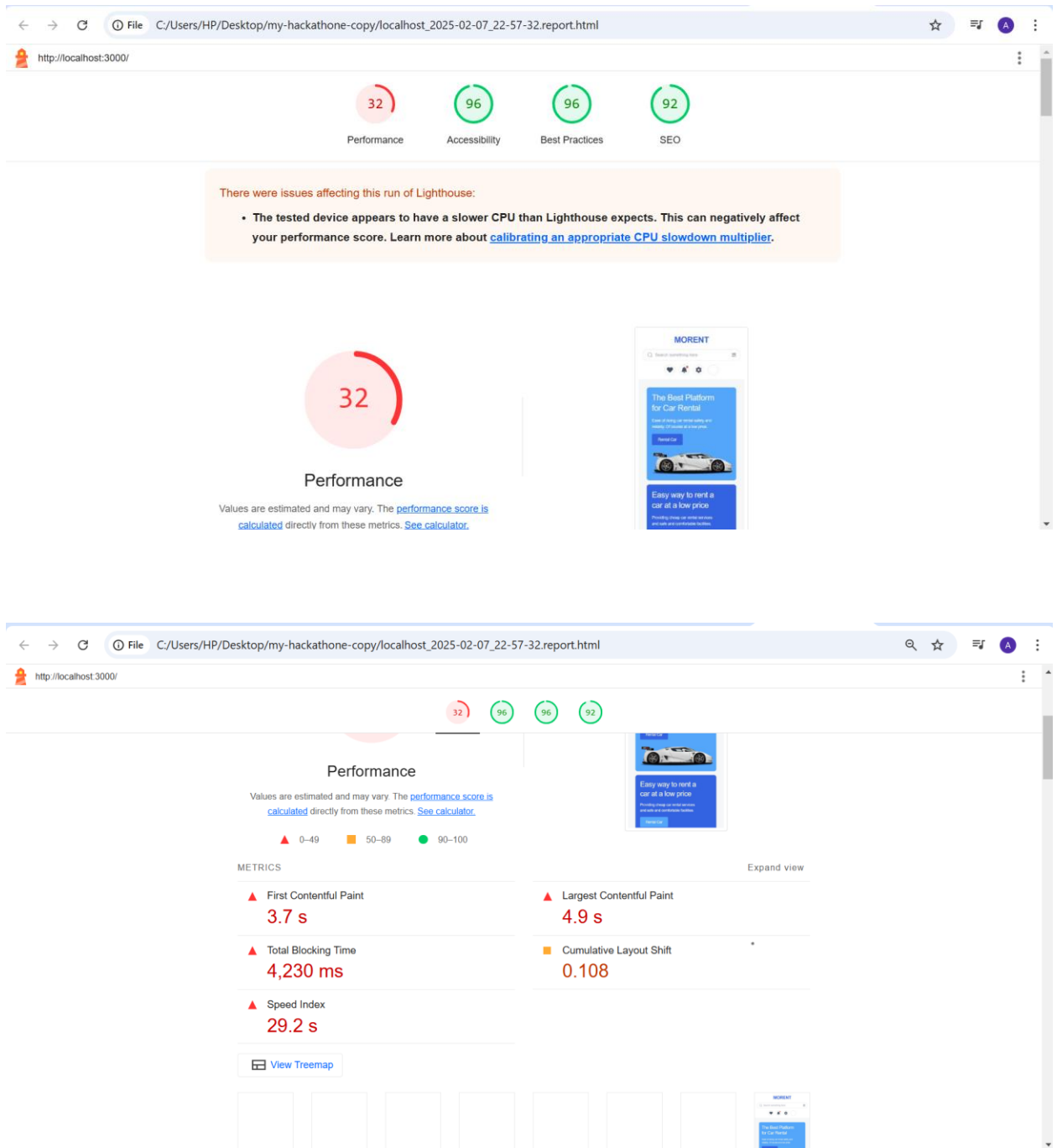
+ Amount

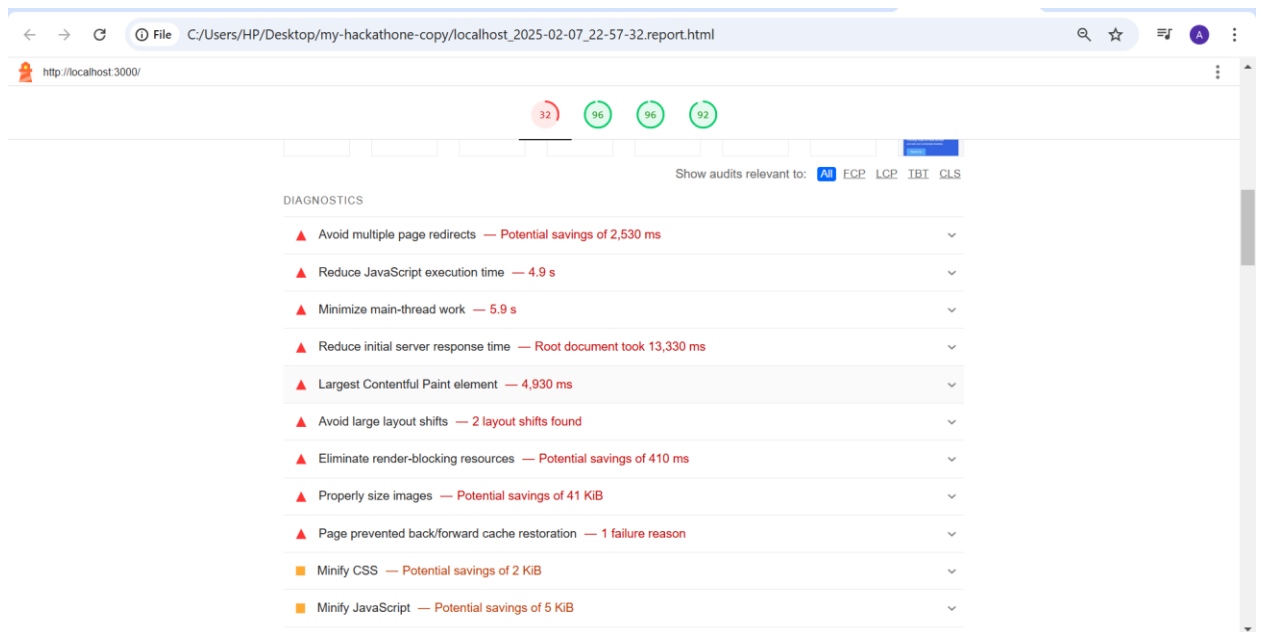
+ Currency

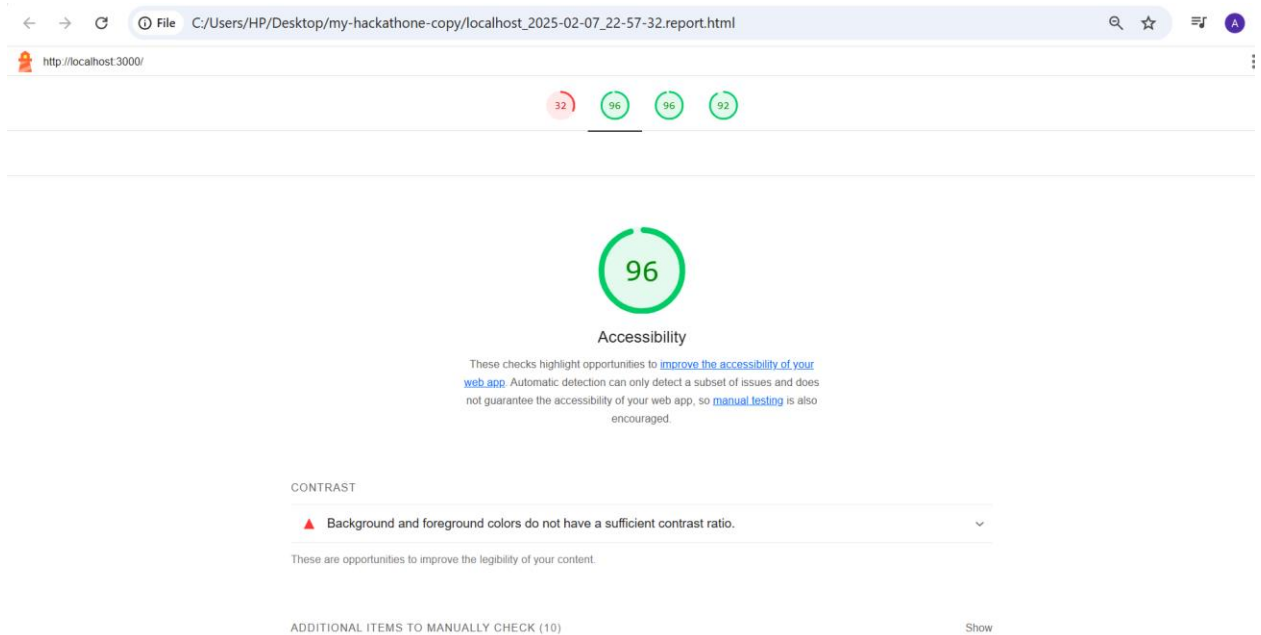
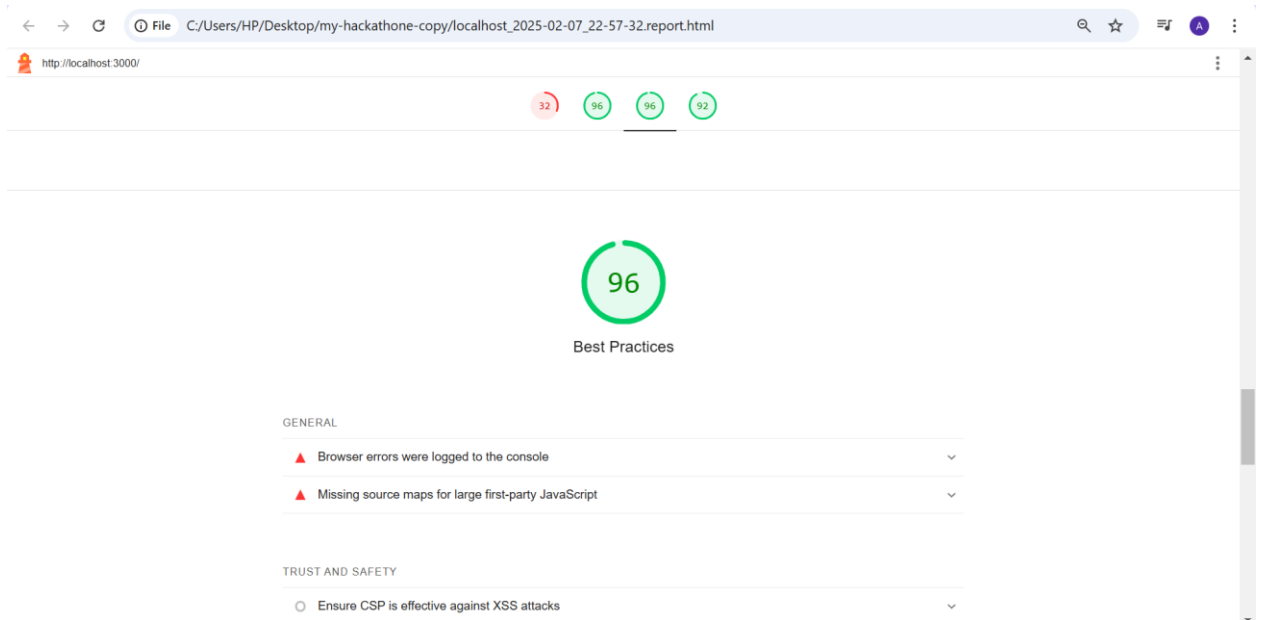
<input type="checkbox"/>	Amount	Payment
<input type="checkbox"/>	\$20.00 USD <div>Succeeded ✓</div>	<div>VISA</div> ...
<input type="checkbox"/>	\$20.00 USD <div>Incomplete ⌚</div>	—
<input type="checkbox"/>	\$20.00 USD <div>Succeeded ✓</div>	<div>VISA</div> ...
<input type="checkbox"/>	\$20.00 USD <div>Incomplete ⌚</div>	—
<input type="checkbox"/>	\$20.00 USD <div>Incomplete ⌚</div>	—
<input type="checkbox"/>	\$20.00 USD <div>Incomplete ⌚</div>	—

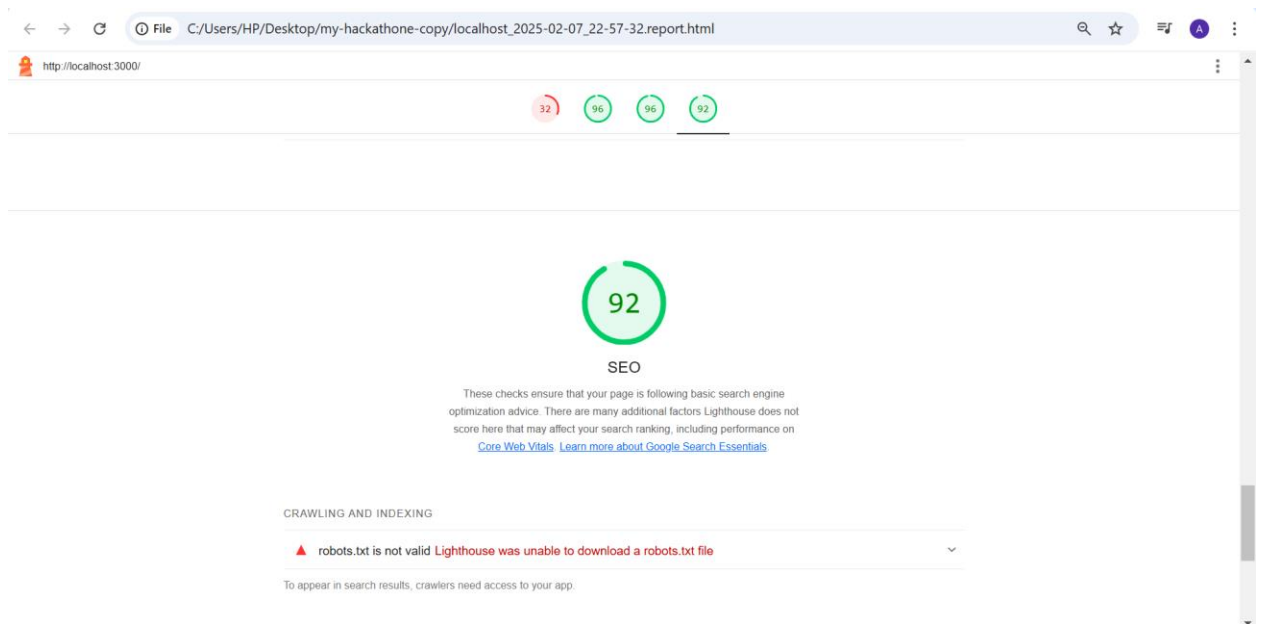
- Errors like insufficient funds or incorrect card details are captured, providing users with appropriate feedback.

5. Performance Optimization:





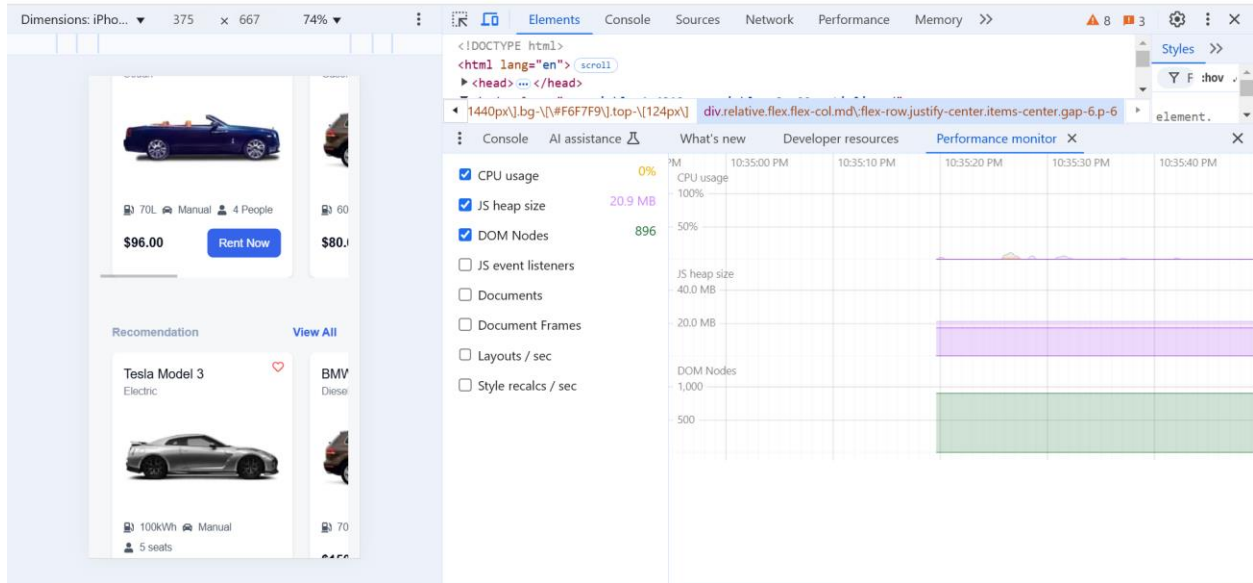




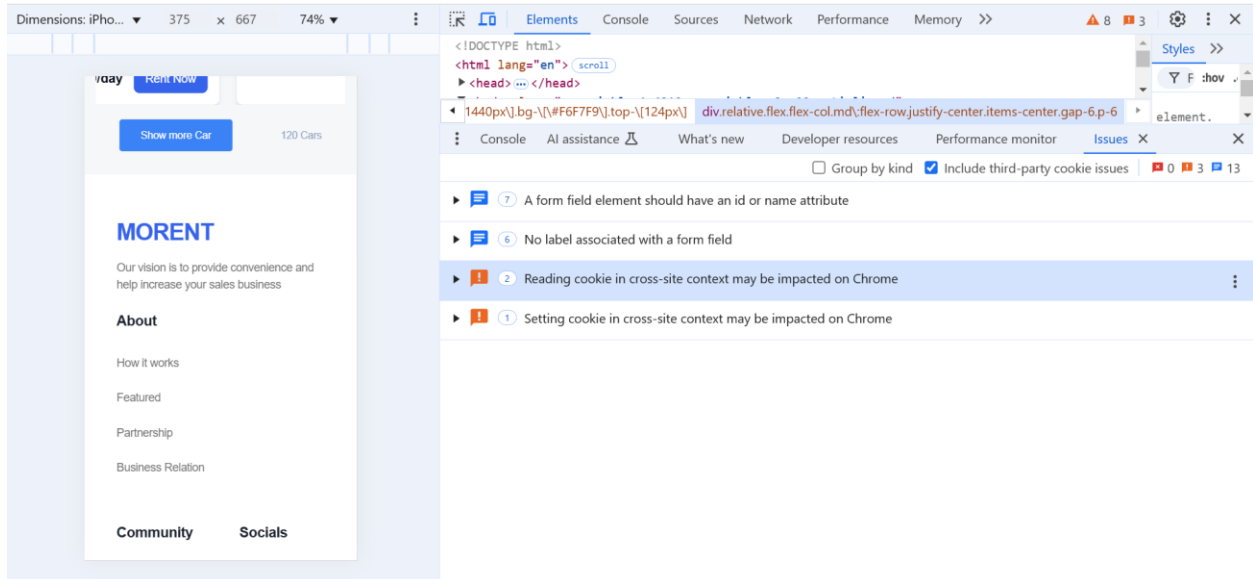
Developer resources

Status	URL	Initiator	Total Bytes	Error
failure	http://localhost:3000.../invalidateCache.js.map	http://local...		HTTP error: status code 404, net:E...
success	/IDxGYUHNXYJ0IC8+IDogPEZhUmVnSGVhcnQ		19 692	
success	/json;charset=utf-8;base64,eyJ2ZXJzaW9uIjoz		529	
success	/json;charset=utf-8;base64,eyJ2ZXJzaW9uIjoz		668	
success	/MTZkYldLCjZb3VyY2VzQ29udGVudCI6WyJl		1 165	
success	/json;charset=utf-8;base64,eyJ2ZXJzaW9uIjoz		501	
success	/json;charset=utf-8;base64,eyJ2ZXJzaW9uIjoz		22 420	
success	/json;charset=utf-8;base64,eyJ2ZXJzaW9uIjoz		293	
success	/json;charset=utf-8;base64,eyJ2ZXJzaW9uIjoz		293	

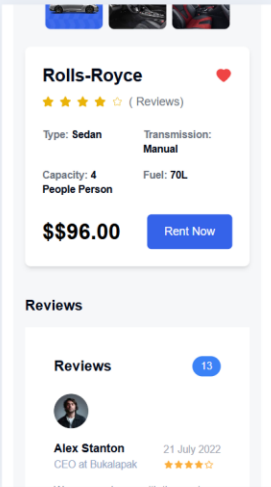
Performance monitor:



Issues:



Security:



Security

- Overview
- ▼ Main origin
- ▼ Secure origins
 - <http://localhost:30...>
 - <https://ob11289y.a...>

Security overview

This page has a non-HTTPS secure origin.