Shnyar Muhamad, N19a

Gamedeals Checker

Short Description:

For this project I used the Svelte Framework and the CheapShark API to make a website. This website, called “Gamedeals Checker”, allows you to search for PC games and view where you can buy them. It helps you to quickly find the cheapest place to buy a specific game and search for games, which are on sale. It works by sending a request to the CheapShark API with all the options that are specified beforehand by the user. The results from that request will then be visible to the user, who can then get more information on a specific deal. The information for the selected deal is gotten by another request to the API.

User Manual:

At the top of the website there are different search options, which help you to narrow down your search. The first search option is a text input, in which you can type in the game of the name which you want to search for. This can be left empty if you don’t want to search for a specific game. The next two options can be used to set a minimum and a maximum price for the deals. This is useful if you are on a budget or want to spend a bigger amount of money on a good game. These two options can of course again be left empty if you don’t have any preferences. The next option allows you to select the stores, which should be included in the search. There are a lot of stores where you can buy PC games, so it helps to narrow down your search to a few stores which you know and trust. By default, 7 out of 35 stores are selected. These 7 stores are stores which I personally know and have used myself. The second last option allows you to set the sorting of the results, which helps if you are looking for something specific. With the last option you can only search for games which are on sale. This again helps if you are on a budget. After setting all your search options, you can then search for the games with the search button.

After the website gets a response back from the API, all the found games will be displayed on the website in 2 columns. For every game a few bits of information will be displayed, including the name, the price, an image of the game and, if available, the rating it has on steam. Every page will display 60 different games. There are buttons at the bottom of the page that allow you to go to the next page or go back to the last one.

If you find a deal you like, you can click on it to get more information. Once you click on it, you will be redirected to a new view. This page now includes all the before mentioned information and some extra information like the game’s Metacritic score, which can be clicked on to get to the game’s

Metacritic page. Additionally, all deals from other stores which are cheaper than the one which is being viewed will be displayed with the possibility to click on them to get more information on them. Lastly, a button exists on the right side, which will redirect you to the store's page, where you can buy the game.

Tutorials and Help used:

Most of the website I made by myself without the help of tutorials. Most of the help I got was from the official Svelte Tutorial [1], which I mostly used when I forgot the exact syntax of a statement, and the CheapShark API Documentation on the RapidAPI [2] and the official Cheapshark [3] website. Of course, I also used websites like Stackoverflow [4] and MDN Web Docs [5], but I only used these for single lines of code and not big parts of my code.

At one point I was stuck during the development of the website, because I couldn't get a proper response from the API when trying to get information about a specific deal. So, I went to the official CheapShark Discord Server [6] where I tried to get help. There I got a response from the creator of the CheapShark API, who assisted me in fixing the problem. The problem was that the ID I used to try to get the information about a specific deal was encoded to URL-encoding. When I then made the request the code would encode the string again, making it double-encoded and therefore not usable by the API. To fix this I simply decode the string, so that the API receives a string which is only encoded once and therefore usable by the API.

But, like mentioned before, most of the work I did by myself, and I did not use a big in-depth tutorial.

Links:

- [1] Svelte, Tutorial, https://svelte.dev/tutorial/basics [Last access: 23.12.22]
- [2] RapidAPI, CheapShark – Game Deals API Documentation, https://rapidapi.com/CheapShark/api/cheapshark-game-deals/ [Last access: 23.12.22]
- [3] CheapShark, Cheapshark API, https://apidocs.cheapshark.com/ [Last access: 23.12.22]
- [4] Stackoverflow, https://stackoverflow.com/ [Last access: 23.12.22]
- [5] MDN Web Docs, https://developer.mozilla.org/en-US/ [Last access: 23.12.22]
- [6] Discord, CheapShark, https://discord.com/invite/cheapshark [Last access: 23.12.22]

Images of the website :