CheapBuy - Extension to fetch cheapest price of the product

Het Patel
hpatel28@ncsu.edu
NC State University,
North Carolina, USA

Hardik Udeshi
hvudeshi@ncsu.edu
NC State University,
North Carolina, USA

Saloni Mahatma
smahatm@ncsu.edu
NC State University,
North Carolina, USA

Kalgee Kotak
kkotak@ncsu.edu
NC State University,
North Carolina, USA

Vineet Chheda
vchheda@ncsu.edu
NC State University,
North Carolina, USA

ABSTRACT

The idea of this document is to summarize the purpose, functionality and the overall design of the extension built as a part of the SE Fall2021 batch project. cheapBuy Extension provides you ease to buy any product through your favourite website's like Amazon, Walmart, Ebay, Bjs, Costco, etc, by providing prices of the same product from all different websites to extension. It takes lot of time to search for the same product in different websites, and find the cheapest one, instead just add our extension cheapBuy and it will automatically fetch you price of the same product from different websites and you can directly compare the prices from different websites through our extension. In sum, cheapBuy is an one stop solution to buy the cheapest product online. The technologies used and future scope is summarized in the following sections.

KEYWORDS

scrapping, Amazon, ebay, walmart, bjs, costco, cheapest price

ACM Reference Format:
Het Patel, Hardik Udeshi, Saloni Mahatma, Kalgee Kotak, and Vineet Chheda. 2021. CheapBuy - Extension to fetch cheapest price of the product. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.5281/zenodo.5540375

1 INTRODUCTION

Manually comparing the prices on different website can be really frustrating. It takes lot of time to find the same product on different website and then buy the cheapest product. Majority of the times people end up buying the product with higher price, as they don't compare the prices from different website. The purpose of this **cheapBuy** extension is to fetch the prices of the same product so that user can buy any product having cheapest rate.

cheapBuy Extension provides you ease to buy any product through your favourite website's like Amazon, Walmart, Ebay, Bjs, Costco, etc, by providing prices of the same product from all different websites to extension. It takes lot of time to search for the same product in different websites, and find the cheapest one, instead this extension cheapBuy will automatically fetch you price of the same product from different websites and you can directly compare the prices from different websites through our extension. In sum, cheapBuy is an one stop solution to buy the cheapest product online.

2 EXTENSION OVERVIEW

As noted in the introduction, the extension built will notify the user about the cheapest rate of the same product across multiple website. When user is on their favourite website to buy any product, they can simply add out extension **cheapBuy** and it will automatically fetch you product name, product price, product link from different websites. Hence, by user can clearly compare the price and make the best and cheapest choice to buy that product.

2.1 Future Scope

Several advance features can be added in the extension such as:

(1) Display badges on the website instead of details in the extension.
(2) Improve accuracy to fetch same product(using NLP techniques).
(3) Recommendation of the cheapest product using colors(i.e. green for cheapest).
(4) Alternate product suggestion feature (if same product not found).
(5) Develop dashboard for extension to monitor activities of user.
(6) Handle server using Terraform.

3 KERNEL BEST PRACTICES FOLLOWED

The overall project was built following the best practices in software development. Below are listed all the kernel best practices and it's implementation with examples in the defined project:

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.5281/zenodo.5540375

3.1 Zero Internal Boundaries

We ensured that everyone in the team has access to all the tools being used. The back-end was coded using Python and Javascript and the team members were comfortable using the same.The front-end of the application was developed using HTML5 that was compatible with the systems used by all team members. An AWS Server was used as a part of infrastructure to make the process easy. Since most languages were known by everyone in the team, we were easily able to communicate and also complete our own pieces.

3.2 Short Release Cycle

The overall release cycle focused on delivering MVP version of the application and hence there were no shorter releases in the window of 4- weeks. Almost all codes were pushed in a span of 2-3 days and a few bug fixes were pushed after a testing cycle.

3.3 Distributed Development Model

The overall work was divided well amongst all team members considering the everyone's schedules. Once the project idea was brainstormed and aligned upon, two of us heavily worked on front-end and the rest three on back-end and documentation. However,everyone was aware of the statuses and progress across all the divisions.

3.4 Consensus Oriented Model

There were several discussions and brainstorm meetings that happened to align on the overall project idea, the outcomes that we wanted to achieve and overall design we wanted to follow. As we progresses through developing the system, all the issues, uncertainties and challenges were discussed, opinions were heard

3.5 No Regression Rule

Currently, the scope does not have any issues that would require regression testing.The future scope however does define various modules / additional functionality that would require regression testing of the complete system.