

CS 3307 HOP

(Shop Without Searching)

Group 8

Maxwell Ding (jding263)
Yulun Feng (yfeng445)
Truman Huang (yhuan939)
Matthew Owen Tjhie (mtjhie)
Huiliang Xia (hxia47)

Table of Content

Description.....	2
Overview.....	2
Software Requirement.....	2
Features.....	3
Required Features.....	3
Optional Features.....	3
Wishlist Features.....	3
Risks.....	3
References.....	4

Description

Overview

In the age of digital transformation, the way users interact with online platforms is undergoing a significant shift. Recognizing this evolution, we introduce HOP, a revolutionist web app premeditated to redefine the online shopping experience.

HOP is not just simply another e-commerce platform; it assists users in finding their desired products without them having to precisely articulate their search. By seamlessly integrating voice recognition, visual recognition, and natural language processing, our advanced platform enables users to effortlessly search for products without relying on traditional text-based queries. Imagine a shopping experience where you simply describe the scenario, requirements, and recipients of the product you're seeking, and the platform handles the rest.

Furthermore, with the text-to-sound feature, HOP ensures that every user, including those with visual impairments, can navigate and interact with the platform with ease. Our SQL/KV database's strong set of rules ensures that the platform can handle large amounts of data and consistently provides accurate and quick outcomes for searches.

One of the standout features of HOP is its keyword tagging system, powered by the OpenAI API. This system ensures that every product is tagged with relevant keywords, making searches more intuitive and results more relevant. Whether you're using voice commands, visual cues, or natural language, our platform understands your needs and delivers results that are tailored just for you.

In essence, HOP is not just about shopping; it's near hopping into an experience where technology understands you, making your online shopping journey smoother, faster, and more enjoyable.

Software Requirement

- C++ as main programming language
- Drogon as web application framework
- MySQL as main database to store our product information
- Redis as KV database to cache user query

- HTML, CSS and Javascript as front-end markup and scripting language

Features

Required Features

- Voice input
- Integrate with OpenAI api to generate summary for each item
- Generate tag from uploaded product pictures using Google Cloud Vision API
- Use browser built-in TTS functionality to give feedback to users
- Provide product suggestion for customer

Optional Features

- Relevance weighting of search engine

Wishlist Features

- Products related to this item.
- Search a product in multiple sites

Risks

One of the risks here is that the product filter, which is responsible for categorising and organising products. To mitigate this risk, continuous monitoring and testing of the product filter should be implemented. Regularly reviewing and updating the filter's algorithms and rules will help ensure that products are accurately categorised. User feedback and input can also be valuable for fine-tuning the filter to improve its accuracy.

Furthermore, there might be a risk that users may provide vague or ambiguous search keywords or descriptions, which could lead to irrelevant search results. One way to resolve this is implementing a keyword ranking system that can help prioritise keywords based on their relevance to the user's query. By assigning weights or scores to keywords, the system can identify which keywords should carry more importance in determining search results.

References

Dale, R. (2021). GPT-3: What's it good for? Natural Language Engineering, 27(1), 113-118. doi:10.1017/S1351324920000601

OpenAI. (n.d.). Whisper: An automatic speech recognition (ASR) system. Retrieved from <https://openai.com/research/whisper>

Gerganov, G. (n.d.). Whisper.cpp. Retrieved from <https://github.com/ggerganov/whisper.cpp/tree/master>

Google Cloud. (n.d.). Cloud Vision API. Retrieved from <https://cloud.google.com/vision>

Drogon Framework. (n.d.). Drogon: A C++14/17 based HTTP application framework. Retrieved from <https://github.com/drogonframework/drogon>

ClickHouse. (n.d.). ClickHouse: A fast open-source OLAP database management system. Retrieved from <https://clickhouse.com/>

OpenAI. (n.d.). OpenAI API Reference. Retrieved from <https://platform.openai.com/docs/api-reference>

MySQL. (n.d.). MySQL: The world's most popular open-source database. Retrieved from <https://www.mysql.com/>

Redis. (n.d.). Redis: An in-memory data structure store. Retrieved from <https://redis.io/>

Google Chrome Developers. (n.d.). Chrome Text-to-Speech API. Retrieved from <https://developer.chrome.com/docs/extensions/reference/tts/>