Aryan Gupta

Middlesex University

Project Report

Drone Price Comparison Website

**Project Report**

**Price Comparison Website**

This website is about price comparison of drones from different websites, which allows users to check the price of a product on a single platform from different websites with their link to it by scraping data from different websites. The website contains multiple Pages(i.e. Home, Product Cards, and Product Page). The website exhibits a Logo at the to bar and a search bar below than where users can search the desired product. The other two pages are the Product Card Page, where the user can view the products cards of searched term, and the Product Display Page, which displays the details and links to different websites of the selected product by the user from the Product Cards Page.

This website contains three pages:

* Home Page
* Product Cards Page
* Product Display Page

**Home Page:**

**A picture containing road

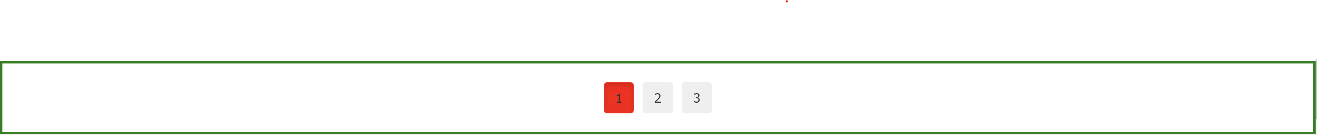
Description automatically generated**

This page exhibits a search bar at the center where the user can search for a product by clicking on the search button, which will take the user to the products card page.

**Product Cards Page:**

A picture containing diagram

Description automatically generated



This page contains the product cards that hold product data and images, by clicking on it the customer will be taken to the product display page. The header has a search bar opposite to logo to search for a new product and a home button visible on this page to return to the home page. At the center of the header, there is an option of sorting products with the name. The pagination number for search term out of total products in the database is being displayed beside sorting option. At the bottom, pagination is being displayed.

**Product Display Page:**

**Graphical user interface, application

Description automatically generated**

This page contains the image and details about the selected product. Besides that, there is a website links section that contains the name and prices of the different websites. The header has a search bar opposite to logo to search for a new product and a home button visible on this page to return to the home page and a back button to get back to the previous page.

**Web Scraping**

The web scraping has been accomplished through selenium by scraping data from different websites, then filtering the data and saving it to the MySQL database using hibernate and dependencies injection using the spring framework. While scraping data sethedless false is used to scrape data otherwise it is not able to scrape data from websites. The data is being stored in a way to avoid any duplication in both the products table and the compared\_url table. Two search term has been used to “drone “ and “dji drone” (DJI a very famous brand) to get a large amount of data. By using two terms there is a chance of link duplication because some links contain the search terms which allow creating duplicate links.

**RESTful Web Services**

The task is accomplished through Express, Node js, and other dependencies.

To get the list of products containing the term dji:

<http://localhost:8080/search/dji>

To get the list of products containing search term dji with a specific limit and offset for pagination:

<http://localhost:8080/search/dji?num_items=2&offset=0>

To get a product with links to a different website through product ID:

<http://localhost:8080/products/productId>

**JavaScript**

Vue.js framework has been used to display products on a single-page web application by fetching data through Axios.

**Testing**

The testing is conducted through Mocha & Chai, to test server and restful services

**Text

Description automatically generated**

The java testing has been done through JUNIT. The testing is conducted by testing scraping classes and productDao class functions.

**Text

Description automatically generated**

**Database:**

**Table

Description automatically generated**

The database includes two tables of products and a compared\_url table.

**SQL Commands:**

This query creates product table with variables. Product\_id is primary and set on auto increment.

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `products` (

`product\_id` **int**(11) **NOT** **NULL** **AUTO\_INCREMENT**,

`name` **varchar**(1000) **CHARACTER** **SET** **utf8mb4** **COLLATE** **utf8mb4\_unicode\_ci** **DEFAULT** **NULL**,

`image\_url` **varchar**(10000) **DEFAULT** **NULL**,

**PRIMARY** **KEY** (`product\_id`)

) **ENGINE**=**InnoDB** **AUTO\_INCREMENT**=34 **DEFAULT** **CHARSET**=**LATIN1**;

This query creates compared\_url table with variables. url\_id is primary and set on auto increment and includes foreign key product\_id from product table.

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `compared\_url` (

`id` **int**(11) **NOT** **NULL** **AUTO\_INCREMENT**,

`product\_id` **int**(11) **NOT** **NULL**,

`url` **varchar**(10000) **CHARACTER** **SET** **utf8mb4** **COLLATE** **utf8mb4\_unicode\_ci** **DEFAULT** **NULL**,

`price` **varchar**(1000) **DEFAULT** **NULL**,

**PRIMARY** **KEY** (`id`),

**KEY** `product\_id` (`product\_id`),

**CONSTRAINT** `compared\_url\_ibfk\_1` **FOREIGN** **KEY** (`product\_id`) **REFERENCES** `products` (`product\_id`)

) **ENGINE**=**InnoDB** **AUTO\_INCREMENT**=46 **DEFAULT** **CHARSET**=**latin1**;

);

**Urls:**

accomplishes by NodeJs, visual studio

Amazon: <https://www.amazon.co.uk/>

Argos: <https://www.argos.co.uk/>

Very: <https://www.very.co.uk/>

Ebay: <https://www.ebay.co.uk/>

Wexphotos: <https://www.wexphotos.co.uk/>

**Tools:**

This project accomplishes by NodeJs, visual studio code, and MySql, Intellij.

**Conclusion:**

I will create a backup of my work on GitHub. I will make sure plagiarism is avoided.