**Dokumentacja Projektu**

**Aplikacja Webowa LCWeb**

Informatyka i Ekonometria

Jakub Sala

Andrzej Podsiadłowski

Mateusz Sęk

**Spis treści**

1. Charakterystyka Oprogramowania
2. Prawa Autorskie I Wymagane Licencje
3. Specyfikacja wymagań
4. Diagram klas
5. Architektura systemu
6. Scenariusze testowania
7. **Charakterystyka Oprogramowania**

Laptop Comperision Website LCWeb – strona webowa, służąca do porównywania laptopów z różnych stron internetowych (EuroRTV i Morele), względem różnych parametrów i filtrów.

Technologia wykorzystana do stworzenia projektu:

* języki: PHP,HTML
* program: Visual Studio Code

1. **Prawa autorskie i Warunki Licencyjne**

Uznanie Autorstwa 4.0 Międzynarodowe (CC by 4.0) – pozwala na kopiowanie, zmienianie, rozprowadzanie, przedstawianie i wykonywanie utworu jedynie pod warunkiem oznaczenia autorstwa.

Projekt stworzony na podstawie tutoriali:

* [https://www.youtube.com](https://www.youtube.com/watch?v=Uc5mfudMTKE)
* [https://www.youtube.com](https://www.youtube.com/watch?v=3wrQHBlmaDo)

1. **Specyfikacje Wymagań**

|  |  |  |  |
| --- | --- | --- | --- |
| **Id** | **Nazwa** | **Opis** | **Priorytet** |
| 1 | Podłączenie adresu URL Euro RTV AGD | By móc swobodnie przechodzić do strony Euro RTV AGD, by następnie korzystać z informacji o produktach i ich specyfikacjach, należy podłączyć adres URL strony | 1 |
| 2 | Podłączenie adresu URL Morele | By móc swobodnie przechodzić do strony Morele, by następnie korzystać z informacji o produktach i ich specyfikacjach, należy podłączyć adres URL strony | 1 |
| 3 | Filtrowanie | By nasza porównywarka posiadała pełną funkcjonalność musi filtrować produkty według upodobań użytkownika, np. cena, pamięć itd. | 2 |
| 4 | Przechodzenie do strony internetowej | Po wybraniu konkretnego produktu, można przejść bezpośrednio do strony, która posiada najlepszą ofertę | 2 |

1. **Diagram klas**

Nasz projekt zawiera jedną klasę o nazwie ScrapperItem, zawiera ona publiczne zmienne.

class ScrapperItem{

public $title;

public $image;

public $price;

public $description;

public $shop\_url;

public $shop;

1. **Architektura systemu**

* Biblioteka PHP Simple HTML DOM Parser Free software pod licencją MIT ( <https://simplehtmldom.sourceforge.io/docs/1.9/> )
* PHP, HTML i Bootstrap ( <https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css> )
* Visual Studio Code
* XAMPP Apache

1. **Scenariusz testowania**

|  |  |  |  |
| --- | --- | --- | --- |
| **Id** | **Wymaganie** | **Opis** | **Wynik** |
| 1 | Scrapper łączy się ze stroną, z której są pobierane dane. | Za pomocą podłączenia adresu URL w kodzie, nasz scraper, ma połączenie ze stroną, którą chcemy scrapować. | pozytywny |
| 2 | Scrapper poprawnie wykorzystuje system filtrowania | Możemy przefiltrować nasze produkty po marce, pamięci RAM, wielkości | pozytywny |
| 3 | Pobieranie zdjęcia ze strony | Kod pobiera zdjęcie przypisane do produktu ze strony | pozytywny |
| 4 | Przechodzenie do strony | Po kliknięciu na wybrany produkt, scraper przenosi nas do strony URL | pozytywny |