

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПАРСИНГУ ВЕБ РЕСУРСІВ ТУРИСТИЧНИХ КОМПАНІЙ

Вінницький національний технічний університет

Анотація

У даній роботі розроблено веб додаток, який складається з парсера актуальних даних туристичних пропозицій та сайту, де вони виводяться.

Ключові слова: *парсинг, веб розробка, фреймворк, односторінковий сайт.*

Abstract

In this article, a web application is developed, which consists of a parser of current data of tourist offers and the site where they are displayed.

Keyword: *parsing, web developing, framework, single page application.*

Вступ

Веб розробка спеціалізується на створенні різних веб інструментів та повноцінних додатків. Існує безліч способів створити найпростіший сайт. Коли вимагається отримати готовий сайт від платформи для побудови сайтів, він зазвичай має набір функцій, яких не можна позбутися. А зайва функціональність створює додаткове навантаження на швидкість сайту.

Веб-розробка на замовлення вирішує цю проблему, даючи вам тільки те, що потрібно — не більше і не менше. Це підвищує швидкість, попереджає конфлікти, прибирає складність і забезпечує безперебійну роботу сайту.

Постановка задачі

Необхідно розробити веб ресурс, сайт-партнер, для видачі актуальних туристичних пропозицій, які необхідно одержати з XML-стрічки.

Сайт повинен бути:

- легкий, щодо функціоналу;
- ефективний, спроможний на обробку великих об'ємів даних;
- швидкий;
- основна ціль – парсинг даних та їх оновлення кожні 15 хв.
- стек технологій: PHP, MySQL, HTML, CSS, JavaScript.

Парсер повинен:

- правильно обробляти вхідну XML ленту з даними;

- оновлювати дані кожні 15 хв, щоб виводити лише актуальні позиції на сайті;
- видаляти не актуальні позиції, або перезаписувати актуальними даними.

Шляхи реалізації

Для вирішення цієї задачі можна використати досить ефективні інструменти, які мають універсальний набір інструментів та зручну архітектуру для побудови складних веб додатків.

Такими інструментами є різні фреймворки, що надають базу для створення потрібного функціоналу та забезпечує високу швидкість обробки даних.

Проаналізувавши задачу, можна прийти до висновків, що на виході необхідно отримати односторінковий сайт у вигляді ленти з великим об'ємом даних. З боку бази даних можна сміло використовувати реляційну БД, тому як буде лише одна таблиця з даними, де усі поля матимуть не нульові значення. Щодо клієнтської частини – усі запити оброблятимуться AJAX інструментом, що надасть зручність у користування односторінковим сайтом.

Для спрощення розробки даного веб-ресурсу та подальшого його масштабування бажано використати фреймворк Laravel.

Парсер даних варто створити, як окремий модуль, що дозволить уникнути зайвих навантажень на сайт та надасть бажану швидкість обробки великої кількості даних. Важливий момент – парсер повинен оновлювати дані з періодом меншим за 15 хв.

Обґрунтування та рішення

З багатьох доступних фремоворків було вибрано Laravel. Він має відносно низький поріг входження, входить у трійку найпопулярніших фреймворків, має розширену документацію та ефективні інструменти. Співтовариство розробників активно впроваджують новий функціонал, що підтримують останні версії PHP.

Фреймворк Laravel дозволяє на базі архітектури MVC побудувати ефективний веб ресурс. Посилаючись на ТЗ, даний фреймворк відповідає основним вимогам, а саме:

- швидка обробка великої кількості даних
- відсутність непотрібного функціоналу, що забезпечує високу швидкість завантаження сайту
- допоміжні інструменти для роботи з БД (Query Builder)
- можливість подальшого масштабування проекту
- використання сучасних інструментів, що залишатимуться актуальними тривалий час
- застосування ООП

Розробка сайту на фреймворці Laravel не сильно відрізняється від інтернет-ресурсів на Symfony і при цьому має ряд переваг:

- широкий і різноманітний функціонал. Можна створювати масштабні інтернет-проекти, незалежно від складності і спрямованості, у тому числі і багаторівневі веб-сайти. Дозволяє красиво вирішувати самі нестандартні завдання;

- можливість тестувати, додавати оновлення, вносити зміни в роботу сайту на додатковій версії. Досягається за рахунок підтримки різних версій інтернет-ресурсу;

- надійний захист бази даних від SQL, CSRF, XSS;

- забезпечує стабільну роботу навіть при великих обсягах даних.

Усі зміни в PHP і напрямів розвитку веб-сайтів обов'язково враховуються в оновленнях початкового коду цього фреймворка і в усіх подальших версіях.

Laravel використовує архітектуру MVC (Model View Controller), що розділяє на окремі логічні модулі структуру сайту. Основний принцип роботи з цією архітектурою – відокремлення 3-х основних частин: відображення контенту (view), обробка запитів та логічна частина сайту (controller) та робота з даними (model). Ці три частини тісно пов'язані між собою.

На базі даного фреймворка можна розробити легку та ефективну архітектуру, де буде всього лише одна модель, декілька контролерів та один view-компонент.

Парсер, як окремий модуль цього веб-додатку, буде реалізований окремим скриптом, що вміщує в себе усі потрібні функції: створення запису в БД, видалення запиту, оновлення запиту. Було проведено аналіз та тест обробки даних. Результати показали, що при простому алгоритмі (цикл з перевітками), скрипт працює не більше 3-х хвилин, що задовільняє рамки, вказані в ТЗ. Для роботи з XML форматом, слід використати розширення для PHP SimpleXML, що дозволяє легко маніпулювати даними одразу ж з файлу.

Висновки

Отже, для вирішення подібних задач, не варто використовувати готові інструменти з не бажаним, зайвим функціоналом. Такий підхід не дасть задовільних результатів щодо ефективності та швидкості обробки даних.

Застосування грамотної бази на основі фреймворків із зручною архітектурою дозволить створити складні, багаторівневі додатки.

Також необхідно відокремлювати парсери та подібні скрипти від сайту, щоб уникнути зайвого навантаження на останній.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Плюси та мінуси розробки сайту на фреймворке Laravel. [Електронний ресурс] – Режим доступу: <https://apri-code.com/uk/razrabotka/plyusy-i-minusy-razrabotki-sajta-na-frejmvorke-laravel/>, вільний.
2. Базовое использование SimpleXML. [Електронний ресурс] – Режим доступу: <https://www.php.net/manual/ru/simplexml.examples-basic.php>, вільний.
3. Что такое MVC. [Електронний ресурс] – Режим доступу: <https://ru.hexlet.io/blog/posts/chto-takoe-mvc-rasskazyvaem-prostyimi-slovami>, вільний.

Артур Васильович Гунько — студент групи ІАКІТ-176, факультет комп'ютерних систем та автоматики, Вінницький національний технічний університет, Вінниця, e-mail: beatport036@gmail.com

Науковий керівник: Паламарчук Євген Анатолійович, кандидат технічних наук, доцент кафедри автоматизації та інтелектуальних інформаційних технологій, Вінницький національний технічний університет, м. Вінниця.

Gunko Arthur V. — Faculty of computer systems and automation, Vinnytsia national technical University, Vinnytsia, e-mail: sinhrofazatronova@gmail.com

Supervisor: **Palamarchuk Yevhen A.**, PhD, Docent of Automatics and Intellectual Informatic Technologies Department, Vinnytsia National Technical University, Vinnytsia city, email: p@vntu.edu.ua