

Іванов Артем,
здобувач першого (бакалаврського) рівня вищої освіти
фізико-математичного факультету
Науковий керівник: Мосіюк Олександр,
кандидат педагогічних наук, доцент,
доцент кафедри комп'ютерних наук та інформаційних технологій,
Житомирський державний університет імені Івана Франка,
м. Житомир, Україна

АНАЛІЗ СТЕКУ ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ ВЕБСЕРВІСУ БІБЛІОТЕКИ

Постановка задачі. Цифровізація процесів стала необхідною складовою сучасного постіндустріального світу. Вона торкається найрізноманітніших аспектів повсякденного і професійного життя. Цьому процесу сприяє ряд факторів, у тому числі масове поширення спеціалізованих вебсервісів, які дозволяють з легкістю отримувати доступ до інформації і послуг для

Секція 4. Технології розробки інформаційних систем

користувачів, а також ефективно керувати цими ресурсами. Активно використовуються відповідні технології і у бібліотечній сфері. Проте значна частина бібліотек та навчальних закладів України не має якісно спроектованого україномовного належного програмного забезпечення, яке відповідало сучасним вимогам та запитам користувачів. Отже постає необхідність створення доступного та легкого в використанні українського аналогу.

У цьому контексті **метою** статті є аналіз й обґрунтування стеку технологій React та Django для розробки масштабованих та високонавантажених інтернет ресурсів на прикладі вебсервісу бібліотеки.

Виклад основного матеріалу. Вебсервіс бібліотеки має забезпечити віддалений доступ зовнішніх користувачів до матеріалів обліку. Окрім цього він повинен бути пристосований до постійного збільшення кількості літератури та обробки великої кількості запитів від відвідувачів.

Щоб визначити набір технологій для розробки відповідного програмного продукту, у відповідності до стандартів, було сформовано список функціональних вимог. Вони передбачали опис функціоналу зовнішніх користувачів, персоналу бібліотеки та користувачів з правами адміністратора.

Для зовнішніх користувачів вебсервісу було передбачено такі дії:

1. Фільтрацію пошуку (за автором, назвою, ключовими словами, роками тощо). В залежності від необхідної точності пошуку, фільтрація може відбуватись за декількома форматами: стандартний, розширений, професійний;
2. Отримання бібліографічного опису літератури (сформованого у відповідності до ДСТУ ГОСТ 7.1:2006);
3. Групування списку літератури за категоріями.

Функціоналом адміністратора є керування роботою інших внутрішніх користувачів (надання паролів, прав доступу тощо). В свою чергу бібліотекари можуть виконувати:

1. Пошук літератури з можливістю фільтрації відповідною до зовнішнього користувача;
2. Генерацію статистики (надходження, вибуття тощо) за звітний період;
3. Експорт картки в форматі .doc (з можливістю попереднього перегляду);
4. Імпорт та експорт літератури (в текстовому або XML форматах);
5. Керувати навчальною базою даних для навчання нового персоналу;
6. Додавати нові примірники літератури.

У відповідності до функціональних вимог були сформовані сценарії користувачів, які дозволили виробити наглядне розуміння структури ресурсу, що представлена у вигляді єдиної інформаційної архітектури програмного рішення (рис. 1).

Секція 4. Технології розробки інформаційних систем

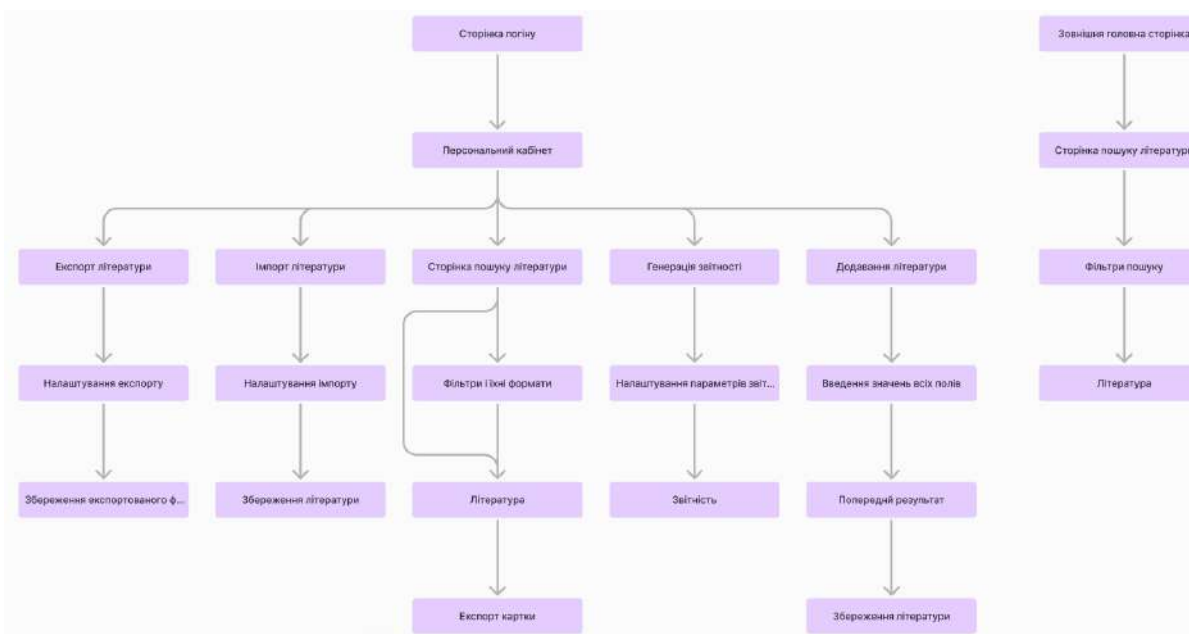


Рис. 1. Інформаційна архітектура вебсервісу бібліотеки.

Виходячи з наведеного рішення, оптимальним для розробки вебсервісу бібліотеки буде використання можливостей мов програмування JavaScript та Python та створених на їх основі технологій:

1. React – бібліотека JavaScript для розробки користувацьких інтерфейсів і клієнтської частини веб-сайтів;

2. Django – фреймворк Python для розробки серверної частини веб-сайтів і програмного інтерфейсів клієнт-серверної взаємодії.

Написання «фронтенду» за допомогою JS-бібліотеки React дає можливість ефективного керування елементами DOM-дерева, а також можливість для подальшого масштабування сайту і видозміни під конкретну організацію.

В свій час, розробка «бекенду» за допомогою Python-фреймворку Django забезпечить високий рівень безпеки ресурсу від ряду розповсюджених атак типу SQL-ін'єкцій і міжсайтових підробок запитів. Орієнтованість Django на шаблон проектування MVT (за яким відбувається розмежування логіки та візуалізації, що забезпечує кращу організацію проєкту та документації) разом з наявністю власної ORM (що спрощує процес взаємодії з базою даних) дозволить пришвидшити та спростити розробку, а модульність додатків Django дозволяє розробляти різну логіку незалежно однієї від одної, а також використовувати вже розроблені рішення в інших проєктах. Підтримка мовою програмування Python асинхронного коду дозволить більш ефективно обробляти велику кількість запитів.

За систему управління базами даних обрано MongoDB, яка є NoSQL СУБД, а саме документо-орієнтованою базою даних, яка за допомогою деревинної структури забезпечує гнучкість при адаптуванні бази даних під стандарти різних організацій. Представляючи документи окремими деревами (структура ліс), ми маємо змогу розмежувати однотомні, багатотомні видання тощо. Завдяки своїй структурі база даних MongoDB є легко масштабованою, а адресація даних дозволяє швидко виконувати запити пошуку в великих об'ємах даних. Варто зазначити, що MongoDB має високу сумісність з Django.

Секція 4. Технології розробки інформаційних систем

Важливою причиною вибору документо-орієнтованої СУБД є неструктурованість даних, що будуть зберігатися. Оскільки ДСТУ ГОСТ 7.1:2006 вимагає збереження в базі даних таких полів, як список ключових слів або зміст (які є множинними і складними полями), набагато ефективніше і зручніше подавати цю інформацію у вигляді JSON документів.

Висновок. Резюмуючи зауважимо, що обрані технології розробки вебсервісу бібліотеки є одними з найкращих і найперспективніших з точки зору як ефективності і швидкодії, так і гнучкості до адаптації до вимог різних організацій. Тобто проаналізований та описаний стек технологій може використовуватись при розробці не тільки бібліотечних застосунків, а й загалом інтернет ресурсів різноманітного призначення, які передбачають опрацювання та зберігання документів.

Список використаних джерел та літератури

1. ДСТУ ГОСТ 7.1:2006 (ГОСТ 7.1–2003, IDT). Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання. – Чинний з 2007–07–01. – К. : Держспоживстандарт України, 2007.
2. Django documentation. URL: <https://docs.djangoproject.com/en/5.1/> (дата звернення: 13.11.2024)
3. Tutorial: Intro to React. URL: <https://legacy.reactjs.org/tutorial/tutorial.html> (дата звернення: 13.11.2024)
4. MongoDB: getting started. URL: <https://www.mongodb.com/docs/atlas/getting-started/> (дата звернення: 13.11.2024)