

ВИКОРИСТАННЯ SPRING FRAMEWORK ДЛЯ РОЗРОБКИ ВЕБ-ДОДАТКІВ

Анотація. Стек технологій Spring Boot, Spring Data JPA, Bootstrap, бібліотека jQuery та база даних PostgreSQL використовуються для розробки веб-додатку. Створений додаток можна використовувати для отримання відповідей на власні питання, допомога у вирішенні чужих питань або пошуку нових знайомих.

Ключові слова: web-додатки, мова Java, Spring Framework, база даних PostgreSQL, MVC шаблон.

Вступ. Однією з сучасніших технологій для створення веб-додатків на платформі Java є Spring Framework [1].

Для розробників програмного забезпечення Spring Framework пропонує дійсно дивовижний інструментарій для розробки веб-додатків та їх налаштування. Це дійсно широка і велика веб-платформа, яка більш ніж здатна взяти на себе будь-яке потенційне завдання або проект.

Spring має модульну структуру. Це дозволяє використовувати тільки ті модулі, які потрібні для конкретного додатку та писати дуже чистий і доступний код.

Spring Framework має досить широку функціональність і активно використовується при розробці складних бізнес-додатків. Spring Framework можна розглядати як колекцію менших фреймворків або фреймворків у фреймворку. Більшість фреймворків може працювати незалежно один від одного, проте, вони забезпечують більшу функціональність при спільному їх використанні[1÷ 4].

Платформа Spring - популярна платформа додатків з відкритим кодом. Вона складається з контейнера, платформи управління елементами і набору інтегрованих служб для веб-інтерфейсів користувача, транзакцій і збереження стану. Також надається підтримка файлів налаштування XML компонентів.

Spring сприяє спрощенню розробки будь-яких додатків на Java, розвантаженню коду; програмний код стає простішим, зв'язок між об'єктами стає більш

слабким; кожен об'єкт займається своєю справою (POJO); використанню принципів ООП на повну потужність; додаткова логіка підключається ззовні; готові вбудовані модулі (робота з базами даних, безпека, транзакції, авторизація, сервіси та інше).

До переваг Spring можна віднести: відносна легкість у вивченні і застосуванні фреймворка в розробці і підтримки додатків; впровадження залежностей (DI) і інверсія управління (IoC) дозволяють писати незалежні один від одного компоненти, що дає переваги в командній розробці, перенесення модулів і тощо; Spring IoC контейнер управляє життєвим циклом Spring Bean і налаштовується зразок JNDI lookup (пошуку); проект Spring містить в собі безліч підпроектів, які зачіпають важливі частини створення софту, такі як Веб-сервіс, веб-програмування, робота з базами даних, завантаження файлів, обробка помилок і багато іншого. Все це налаштовується в єдиному форматі і спрощує підтримку додатків.

На даний час існує безліч додатків, за допомогою яких можна згенерувати резюме за власними бажаннями з різними шаблонами, але, нажаль, вони бувають не дуже актуальними та безкоштовними. В основній масі вони надають дуже не професійний спосіб для заповнення даних, це є проблемою для людей у різних сферах діяльності. Саме резюме є візитною карткою для будь-якої людини під час пошуку роботи або для пошуку однодумців.

Мета роботи знайти і розробити найбільш прийнятні умови для створення резюме, а саме, готовий патерн полів для заповнення. Людина, яка проглядає резюме змогла б точно, швидко оцінити саме ту інформацію, яка для неї є корисною та актуальною. Також, користувачі зможуть переглядати резюме інших користувачів, порівнюючи свої здібності або знання, навички, котрі необхідно здобути та багато іншого.

Мета статті – за допомогою технології Spring розробити веб-додаток для створення форуму програмістів.

Основна частина. Для створення такого веб-додатку було проаналізовано існуючі рішення, тобто, які етапи проходить користувач для створення свого власного профілю на сайті. Слід приділити увагу оформленню профілю для його презентабельного вигляду та можливості редагування тощо. Також проаналізовано недоліки у існуючих рішеннях, щоб уникнути їх.

Одним із найважливіших завдань при розробці програмних продуктів є вибір таких засобів програмування, які зможуть полегшити роботу розробника.

Вони надають всі необхідні інструменти для реалізації поставленого завдання, та дозволяють отримати результат, який задовольняє потребам користувача.

Створення веб-додатку виконувалося на мові програмування Java в середовищі розробки IntelliJ IDEA 2022. При цьому застосовувався стек технологій фреймворк Spring Boot[6]; Spring Data JPA [4]; бібліотека jQuery; фреймворк Bootstrap[5]; база даних PostgreSQL 12[3] та pgAdmin4 (рис. 1).

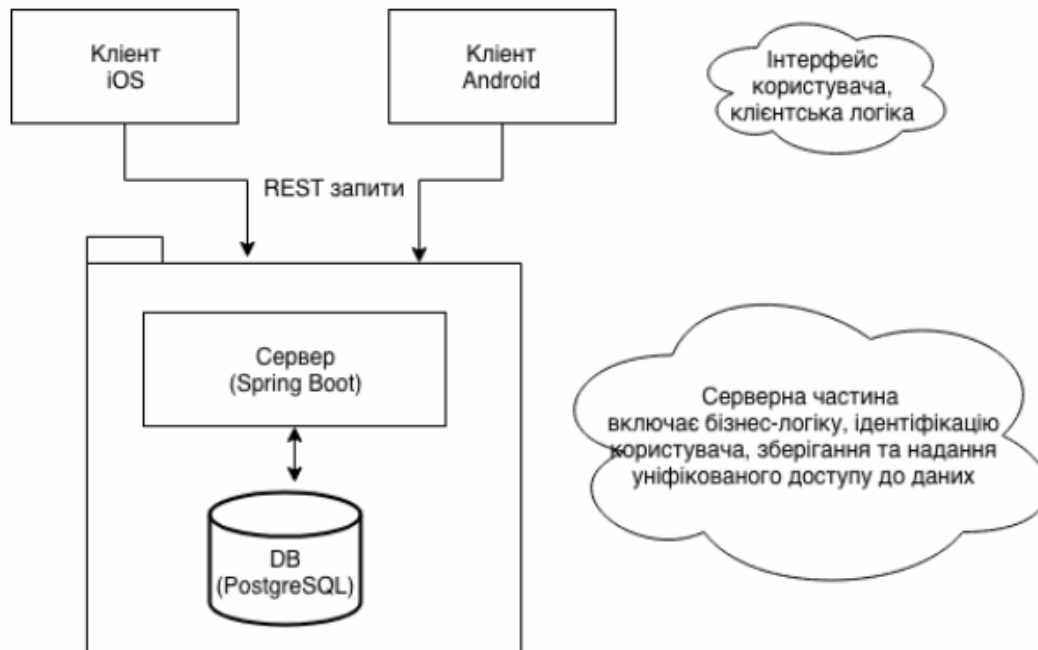


Рисунок 1 – Засоби реалізації веб-додатку

Вибір даних технологій зумовлений тим, що саме ці інструменти у цілому надають найбільшу функціональність серед інших.

Також у проекті застосовано шаблон проектування MVC (model-view-controller), який являє собою архітектурний шаблон модель-представлення-контролер, та може використовуватись під час розробки або проектування додатків.

За архітектурною особливістю такого шаблону систему можна поділити на три взаємопов'язаних частини, одна з них - це модель даних, друга - це вигляд, що представляє собою інтерфейс користувача і третє – це модуль керування.

Метою шаблону є його гнучкий дизайн для додатку, який полегшує подальші зміни або розширення програм, також надається можливість повторного використання окремих компонентів програми. Використання MVC шаблону у великих системах сприяє деякій впорядкованості їх структури та робить більш зрозумілими за рахунок зменшення їх складності.

Модель відповідає за збереження даних та їх структурування, вигляд відповідає за представлення цих даних для користувача, а контролер описує, як саме потрібно керувати компонентами при отриманні сигналів від подій на формі, на веб-сторінці, від дій користувача тощо.

Користувач може створити профіль, який ідентифікується у системі за його роллю, кожен користувач може створити собі профіль, котрий у свою чергу включає таку інформацію, як: основні відомості про себе, короткі дані про навчання, досвід, майстерність, інтереси, свої досягнення.

Для створення програмного продукту розроблена реляційна база даних, яка поєднує усі суб'єкти зв'язками. Всього створено 9 зв'язків та 10 сутностей. Для реалізації цих сутностей створювалась база даних з такими таблицями: **Role** – описує роль користувача у веб-додатку; **User** – зберігає дані профілю користувача; **Address** – зберігає адресні дані користувача; **Experience** – надає дані про досвід; **Education** – дані про освіту; **Degree** – надає для таблиці Education ступені освіти; **Skill** – дані про майстерність; **Interests** – надає дані про інтереси; **About** – надає цікаву інформацію про користувача; **Awards** – інформація щодо досягнень користувача.

Такий стек конфігурацій задовольняє потребам для створення веб-додатку. Весь стек технологій можна стисло описати наступним чином. Завдяки збирачу Maven [3, 7] зчитуються конфігурації (залежності), які знаходяться у pom.xml файлі. Після чого наші view моделі, які знаходяться у каталозі «templates» і являють собою xhtml сторінки (їх безпосередньо бачить клієнт у браузері), можуть взаємодіяти з контролерами, в яких описані певні маршрути та їх параметри. Ці контролери у свою чергу взаємодіють з сервісами, які в свою чергу взаємодіють з репозиторіями, які вже працюють з базою даних. Маємо опис всіх сутностей бази даних – це їх моделі. Кожна модель являє собою опис сутності бази даних та їх зв'язків. З першого погляду кожна така модель схожа на звичайний java клас, але завдяки анотаціям (@Table, @Entity та інші) можна одразу зрозуміти, що це характеризує опис сутності.

Програмний додаток складається з декількох веб-сторінок, особливістю кожної з них є реалізація певного функціоналу та обмеження за правами доступу.

Не зареєстрований користувач має найменші правами доступу. На головній сторінці він може провести пошук за назвою топіку або за автором – це допомагає прискорити пошук. Не авторизований користувач може переглядати профілі інших користувачів та ознайомитись зі вмістом профілю. Саме в про-

філі винесені основні критерії, які можуть бути дуже корисними для надання своєї інформації, а саме: інформація про себе, досвід, навчання, навички, інтереси, нагороди. Останній пункт – це зворотній зв'язок, де можна зв'язатись з обраною людиною через поштовий сервіс .

Якщо користувач зареєстрований, то йому пропонується ввести свої логін та пароль для входу (рис. 2), або ж перейти за посиланням «Зареєструватися». Для реєстрації свого профілю необхідно заповнити всі існуючі поля. При наявності помилок з'являються впливаючі вікна, котрі вказують на помилки при заповненні даних.

Авторизувавшись у системі, користувач потрапляє до головної сторінки веб-додатку. При переході до вкладки «Топіки», користувачу пропонується переглянути список топіків, здійснити пошук за авторами або за назвою та створити своє власний топік, як показано на рисунку 2.

При переході за посиланням «Створити топік» користувач має змогу створити власний топік у відповідності зі своїми інтересами або питаннями для знаходження відповідей.

Назва	Ім'я Прізвище користувача
How to create CRUD project	Solomia Halak
How to configure rights	Solomia Halak

Рисунок 2 – Вкладка авторизованого користувача

Власник профілю має змогу редагувати, доповнювати або видаляти необхідні дані у ньому. Для оновлення поточної інформації необхідно натиснути «Редагувати інформацію» і користувач одразу перейде до іншої сторінки, де матиме змогу редагувати свою інформацію.

Розглянемо функціонал для адміністратора веб-додатку. Адміністратор додатку може бути лише один у системі з відповідними правами доступу. Він має змогу побачити весь додаток, інших користувачів без додаткових прав та додаткові вкладки для адміністрування. Наприклад вкладка «Користувачі» надає змогу адміністратору переглянути повний список користувачів, видалити обліковий запис зареєстрованого користувача або заблокувати його, а також додати нового користувача до системи.

При переході до вкладки «Топіки» адміністратор може знайти необхідний топик за назвою або іменем користувача та переглянути його, якщо воно не відповідає деяким критеріям видалити його. Слід зазначити, що користувач з правами доступу адміністратора не може редагувати свій профіль.

Висновки. Для реалізації функціонуючої системи виконувалися наступні кроки: реалізовано логіку для роботи веб-додатку; створено базу даних для зберігання інформації; реалізовано мінімально приємний дизайн веб-додатку для звичайного користувача у desktop та mobile браузерях; налагоджено багаторівневий доступ до веб-додатку; відокремлено багаторівневий функціонал щодо користувачів додатку з різними правами доступу.

Для подальшого покращення використання та функціонування додатку можна додати приєднання соціальних мереж до профілю, розширене встановлення контакту з людиною через веб-додаток та багато іншого.

ЛІТЕРАТУРА/REFERENCES

1. Spring Framework. [Electronic resource]. Access mode: <https://habr.com/ru/post/490586/>.
2. Maven. [Electronic resource]. Access mode <https://habr.com/ru/post/78252/>.
3. PostgreSQL. [Electronic resource]. Access mode: <https://www.postgresql.org/docs/current/history.html>.
4. Spring Data JPA. [Electronic resource]. Access mode: <https://www.baeldung.com/learn-spring-data-jpa-course>.
5. Mayur Ramgir «Full Stack Java Development with Spring MVC, Hibernate, jQuery, and Bootstrap». Wiley, 2020. – 712 с.
6. Spring Boot. [Electronic resource]. Access mode: <https://spring.io/projects/spring-boot>.

Received 11.01.2023.
Accepted 16.01.2023.

Using Spring Framework to develop web applications

Spring Framework is a popular platform for creating web applications. It consists of a container, an element management platform, and a set of integrated services for web user interfaces, transactions, and state retention. Spring has a modular structure. This allows you to use only those modules that are needed for a specific application and write clean and accessible code.

The purpose of the work is to find and develop the most acceptable conditions for creating a forum of programmers. Users will be able to view profiles of other users, topics, answer questions or create their own.

The web application was created using the Java programming language in the IntelliJ IDEA 2022 development environment. The Spring Boot framework technology stack was used; Spring Data JPA; jQuery library; Bootstrap framework; database PostgreSQL 12 and pgAdmin4.

The project uses the MVC (model-view-controller) design pattern. According to the architectural feature of such a template, the system can be divided into three interconnected parts, the first is a data model, the second is a view representing the user interface, and the third is a control module.

To create a software product, a relational database was developed, which connects all subjects with connections.

The software application consists of several web pages, the feature each of them is the implementation of certain functionality and restrictions on access rights.

Матвєєва Наталія Олександрівна – к. т.н., доцент, доцент кафедри електронних обчислювальних машин, Дніпровський національний університет імені Олеся Гончара

Мохов Кирило Володимирович – магістр за спеціальністю «Компютерна інженерія», Дніпровський національний університет імені Олеся Гончара

Matveeva Nataliya – Candidate of Technical Sciences, associate professor, associate professor of the department of Electronic Computing Machinery, Oles Honchar Dnipro National University.

Mokhov Kurulo - Master of Science in Computer Engineering, Oles Honchar Dnipro National University.