

СЕКЦІЯ «МАТЕМАТИЧНЕ І ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ»

Марта-Діана Басішин, Ольга Мокрицька*

¹Кафедра САП, Національний університет «Львівська політехніка», Україна, Львів, вул. С. Бандери 12, <https://orcid.org/0009-0006-1114-0721>
marta-diana.basishyn.pp.2022@lpnu.ua

²Кафедра САП, Національний університет «Львівська політехніка», Україна, Львів, вул. С. Бандери 12, <https://orcid.org/0000-0002-2887-9585>
olha.v.mokrytska@lpnu.ua

Розроблення інтелектуального веб-застосунку управління завданнями на основі Java Spring Boot

***Анотація.** У роботі розглянуто створення веб-застосунку ToDo List на основі фреймворку Spring Boot із впровадженням елементів інтелектуального аналізу логів. Показано архітектуру системи за моделлю MVC, механізми захисту користувацьких даних із використанням Spring Security та можливості розширення системи за рахунок модулів аналітики. Запропоновано концепцію інтелектуального моніторингу журналів подій на основі карти Кохонена (Self-Organizing Map, SOM), яка дозволяє виявляти аномальні шаблони поведінки в роботі застосунку. Визначено, що поєднання архітектурної гнучкості Spring Boot і алгоритмів штучного інтелекту забезпечує високу надійність і масштабованість застосунку.*

***Ключові слова** – Java, Spring Boot, MVC, логування, безпека, SOM, аномалії, штучний інтелект.*

Сучасні веб-застосунки потребують не лише високої продуктивності та зручності, але й здатності до самоаналізу. Зі збільшенням кількості користувачів і складності архітектури виникає потреба у постійному моніторингу поведінки системи в реальному часі. Одним із напрямів розвитку таких застосунків є впровадження елементів інтелектуального аналізу логів, що дозволяє автоматично виявляти потенційні відхилення у роботі без втручання розробника. Метою цієї роботи є створення веб-застосунку ToDo List на платформі Spring Boot з архітектурою MVC, реалізацією модулів безпеки та розширенням у вигляді концептуального AI-модуля виявлення аномалій у логах.

Розроблений застосунок побудовано за шаблоном Model-View-Controller, який забезпечує чітке розділення логіки на рівні моделі, представлення та контролера. Це дозволяє спростити підтримку, масштабування та повторне використання коду. На рівні даних використано Spring Data JPA у поєднанні з ORM Hibernate, що забезпечує ефективну взаємодію з базою даних без необхідності ручного написання SQL-запитів. У межах бізнес-логіки реалізовано CRUD-операції для створення, редагування, видалення й перегляду завдань користувача.

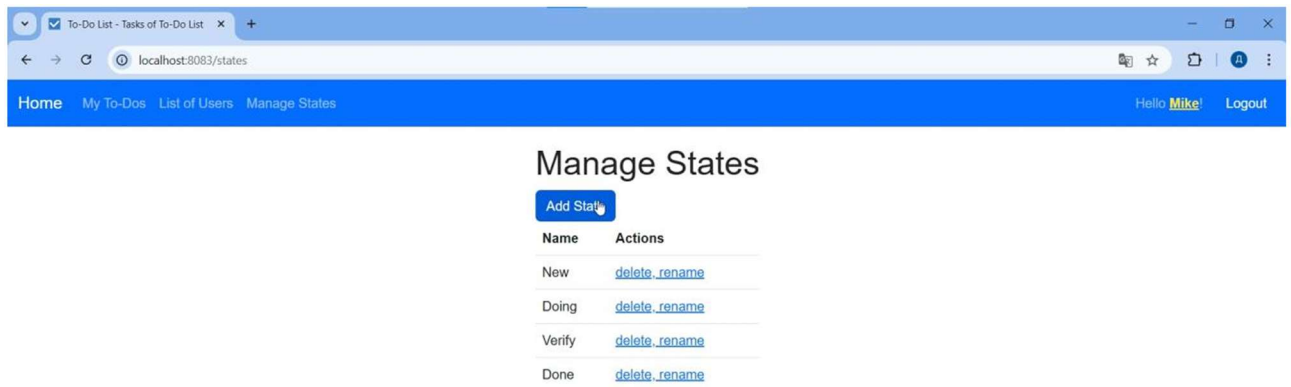


Рисунок 1. Інтерфейс веб-застосунку ToDo List для керування станами завдань

Безпеку застосунку забезпечено через фреймворк Spring Security, який реалізує автентифікацію та авторизацію користувачів із розмежуванням прав доступу за ролями USER та ADMIN. Для захисту паролів використано алгоритм шифрування BCrypt. Такий підхід гарантує надійність зберігання даних і відповідає сучасним вимогам до веб-безпеки.

Для перевірки функціональності застосунку проведено модульне та інтеграційне тестування з використанням JUnit 5, Mockito та MockMvc. Тестування дало змогу підтвердити коректність реалізованих CRUD-операцій, стабільність обробки запитів і стійкість веб-застосунку до некоректного введення даних. Журнали подій формуються за допомогою бібліотек SLF4J та Logback у структурованому форматі, що дозволяє фіксувати ключові параметри запитів – час виконання, статус відповіді, IP-адресу користувача та інформацію про можливі винятки.

Особливу увагу в роботі приділено концепції інтелектуального аналізу логів. Для підвищення надійності веб-застосунку запропоновано реалізувати додатковий модуль, який автоматично виявлятиме нетипові події, здатні свідчити про потенційні помилки або проблеми з продуктивністю. Одним із перспективних методів для цього є Self-Organizing Map (SOM) – нейронна мережа, що здійснює кластеризацію даних за ступенем їх схожості. SOM формує двовимірну карту, де кожен вузол відображає певний тип поведінки веб-застосунку: звичайні запити, помилки, тривалі затримки у відповіді сервера або підозрілу активність користувачів. “Гарячі” області карти відповідають типовим подіям, тоді як ізольовані вузли вказують на рідкісні або потенційно аномальні ситуації.

Такий підхід не потребує безпосередньої інтеграції з основною системою і може реалізовуватись як окремий аналітичний компонент, який періодично аналізує журнали подій, зібрані за певний період. Отримані результати можуть бути подані у вигляді теплових карт (U-Matrix), що спрощує візуальну оцінку стану застосунку та виявлення аномалій. Впровадження подібних інтелектуальних рішень у процес моніторингу дозволяє своєчасно попереджати критичні ситуації, знижувати ризик відмови програмного забезпечення та підвищувати рівень кібербезпеки.

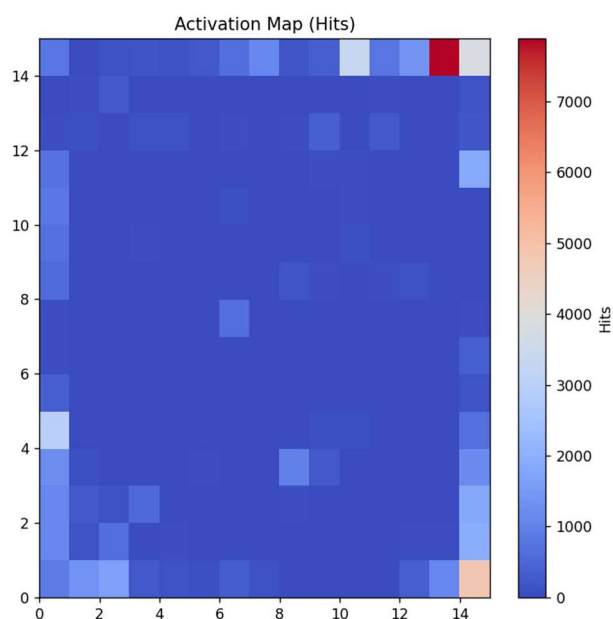


Рисунок 2. Карта активацій SOM для аналізу логів застосунку

Подальший розвиток веб-застосунку передбачає створення веб-панелі моніторингу, яка відображатиме результати аналітики у реальному часі, а також інтеграцію з DevOps-інструментами, зокрема Grafana та ELK-Stack, для централізованого контролю стану застосунку. Перспективним напрямом є також використання рекурентних нейронних мереж (RNN, LSTM) для прогнозування навантаження та виявлення поведінкових трендів користувачів.

Висновок. Отже, розроблений веб-застосунок ToDo List на базі Spring Boot демонструє можливість поєднання класичної архітектури корпоративних систем із сучасними методами штучного інтелекту. Такий симбіоз забезпечує гнучкість, масштабованість і надійність програмного продукту, а також формує основу для створення “розумних” інформаційних систем, здатних до самодіагностики, аналітики та адаптації до зміни умов експлуатації.

Список використаних літературних джерел

- [1] Spring Framework Documentation [Електронний ресурс] – <https://spring.io/projects/spring-framework>
- [2] Spring Boot Reference Guide [Електронний ресурс] – <https://docs.spring.io/spring-boot>
- [3] Spring Security Reference [Електронний ресурс] – <https://docs.spring.io/spring-security>
- [4] Kohonen T. *Self-Organizing Maps*. Springer, 2001.
- [5] SLF4J / Logback Documentation [Електронний ресурс] – <http://logback.qos.ch>
- [6] Heaton J. *Artificial Intelligence for Humans. Volume 1: Fundamental Algorithms*. Heaton Research, 2020.

Development of an Intelligent Task Management Web Application Based on Java Spring Boot

Marta-Diana Basishyn, Olha Mokrytska

The paper presents the development of an intelligent ToDo List web application built on the Spring Boot framework with integrated log analytics. The system architecture based on the MVC pattern, secure user authentication with