

# Курсова робота

Тема: Система зберігання та пошуку дипломних та курсових робіт


Виконав: Добровольський І. С.

# Що було зроблено?

У процесі виконання курсової роботи, було створено прототип системи для зберігання дипломних та курсових робіт.



Works




Requests



Система зберігання та пошуку дипломних та курсових робіт





[Курсова робота.docx](#)



Title	File	User	
Система зберігання та пошуку дипломних та курсових робіт	<a href="#">Курсова робота.docx</a>	i.dobrovolskyi@ukma.edu.ua Добровольський	  

Items per page: 10

1 – 1 of 1



# Система зберігання та пошуку дипломних та курсових робіт

- Було розроблено веб-застосунок, що включає серверну частину та веб-сторінку.
- Користувачі можуть завантажувати, переглядати роботи та здійснювати пошук за запитом.
- Розроблено механізм запитів на створення роботи: користувач створює запит, а адміністратор перевіряє роботу та підтверджує або скасовує запит на створення.
- Автентифікація та авторизація відбувається через сервіс Azure Active Directory.

# Серверна частина

- Серверна частина застосунку написана мовою програмування Java з використанням Spring Framework.
- Spring Boot надає можливість швидко розгортати веб-сервер як локально, так і в хмарі.
- Для інтеграції з Elasticsearch було використано так званий Spring Starter.
- Автентифікація та авторизація реалізовані з використанням Spring Security.

# Реалізація пошуку

- Для реалізації ефективного пошуку робіт за вмістом було використано Elasticsearch.
- Elasticsearch надає можливість зберігати, індексувати документи та здійснювати пошук за запитом.
- Взаємодія між серверною частиною та Elasticsearch відбувається через протокол HTTP.

# Обробка файлів різних форматів

- Індексація файлів різних форматів (наприклад, pdf чи doc) реалізована з використанням плагіна “attachment-pipeline”.
- Для роботи плагіна “attachment-pipeline” необхідно створити так званий “pipeline”, який вказує, які саме поля документу потрібно індексувати з його використанням.

# Веб-сторінка

- Для розробки веб-сторінки було використано фреймворк Angular.
- Angular надає готові UI компоненти: кнопки, поля для вводу, діалогові вікна тощо.
- Автентифікація та авторизація реалізовані з використанням бібліотеки “microsoft-adal-angular6”.
- Неавторизований користувач перенаправляється на сторінку для вводу імені користувача та паролю. Введені дані надсилаються на сервер Azure Active Directory, який повертає спеціальний ключ.
- Згенерований ключ надсилається на сервер разом з кожним запитом, оскільки запити без правильного ключа сервер відхиляє.