Wojskowa Akademia Techniczna



***Programowanie Zdarzeniowe***

**PROJEKT**

Autor:

st. szer. Pchor. Michał LECH

Grupa: I6B1S1

Spis treści

[1. Wstęp 3](#_Toc728539)

[2. Wybór środowiska oraz technologii 4](#_Toc728540)

[3. Przypadki użycia 5](#_Toc728541)

[4. Architektura aplikacji 6](#_Toc728542)

[5. Testowanie 7](#_Toc728543)

[6. Zakończenie 7](#_Toc728544)

# Wstęp

Dokumentacja dotyczy projektu implementacji serwisu który realizuje zadania operacji na plikach cyfrowych.

Dokumentacja projektowa jest równocześnie sprawozdaniem z zadania projektowego jak i dokumentacją techniczną wytworzonego oprogramowania.

W rozdziale drugim zostały przedstawione i uzasadnione wybory na poziomie planowania, takie jak wybór technologii, środowiska, rozwiązań itp.

Następne rozdziały (trzeci, czwarty) dotyczą modelowania projektu i opisują projekt z kolejno następujących perspektyw:

* funkcjonalnej (przypadki użycia);
* architektury;
* procesowej.

Rozdział piąty poświęcony jest testowaniu aplikacji. Projekt nie zawiera testów jednostkowych, ale zostanie zaprezentowany scenariusze w jaki sposób testowana była aplikacja, oraz ich rezultaty.

W rozdziale szóstym są wysunięte wnioski z projektu, oraz zaprezentowane możliwe rozwinięcia projektu.

# Wybór środowiska oraz technologii

Aplikacja została zaimplementowana w oparciu o platformę JavaEE(Java Enterprise Edition). Jest to platforma pozwalająca tworzyć aplikacje biznesowe i zawierająca zestaw specyficznych dla języka Java narzędzi, które upraszczając rozwiązywanie złożonych, skomplikowanych ale jednocześnie powtarzalnych problemów związanych z wytwarzaniem oprogramowania. Java EE wspiera takie technologie jak:

* Servlet/Filter
* JSP/JSTL
* JSF
* JMS
* JPA
* Hibernate
* EJB
* JaxRS

Środowisko w którym została zbudowana aplikacja to Intelllij IDEA która jednocześnie jest naturalnym wyborem kiedy mówimy o języku Java. Naturalnie wspiera ono JEE oraz ułatwia bardzo prosty sposób na komunikacje z wybranym serwerem czy bazą danych. Używa wersja to JDK 1.8. Graficzny interfejs został zaimplementowany w oparciu o HTML5 oraz Java Script.

# Przypadki użycia



**Autoryzacja** – tylko na podstawie poprawnej walidacji danych wejściowych (login i hasło) użytkownik ma możliwość skorzystać z funkcji aplikacji.

**Dodaj plik –** użytkownik wskazuje plik znajdujący się na lokalnej maszynie a następnie serwlet realizujący usługę dodawania zamieszcza go na serwerze.

**Usuń plik –** użytkownik wybierając odpowiedni zasób ma możliwość usunąć zasób cyfrowy.

**Pobierz plik** – użytkownik wybierając odpowiedni plik z listy na serwerze ma możliwość pobrać zasób cyfrowy.

**Znajdź plik** – użytkownik podając dowolny ciąg znaków może wyszukać pliki na serwerze. Wyszukiwanie działa w oparciu o rozszerzenia, nazwy czy tez opis plików.

# Architektura aplikacji

Jak wcześniej określono aplikacja korzysta z Java JEE oraz dodatkowo komunikuje się z lokalną bazą danych H2. Jest to bardzo ciekawa o ile nie jedna z lepszych lokalnych baz danych ponieważ:

- napisana w czystym języku Java

- open source

- wspiera SQL i JDBC API

- posiada embeded(lokalne połączenie przez JDBC), serwer, mixed mode

- kodowanie danych

- kompresje do CLOB/BLOB

- bardzo lekka, około 1,5MB

Za komunikacje miedzy aplikacja a bazą danych został podpięty Hibernate. Architektura aplikacji przedstawiona jest na rysunku poniżej.



# Testowanie

Ponieważ aplikacja nie jest dużym serwisem i prawdopodobnie nie będzie rozwijana zdecydowano się w całości przetestować system manualnie. To co zostało sprawdzone to wszystkie dostępne formularze oraz walidacja danych wejściowych. Dobrze wykonane testy dowodzą, że stworzone serwlety realizują swoje funkcje w sposób prawidłowy. Jednocześnie można potwierdzić, że komunikacja na drodze aplikacja-baza danych również działa prawidłowo. Proces operacji na danych przebiega zgodnie z założeniami. Nie zdecydowano się na testy niefunkcjonalne. Ponieważ aplikacja działała na maszynie lokalnej trudnym zadaniem było przetestowanie jej pod kątem wydajnościowym oraz bezpieczeństwa. Jednocześnie warto zauważyć, ze dobrym podejściem jest zawsze pisanie jednostkowych testów automatycznych, które pozwalają ograniczyć manualny nakład pracy. Podsumowując podczas testów nie znaleziono żadnych rażących błędów czy też błędów które uniemożliwiałby prawidłowe działanie aplikacji.

# Zakończenie

Stworzona aplikacja mimo prostoty swojego działania pokazuje w jak prosty sposób są realizowane aplikacje w technologii JEE. Ponieważ sam projekt jest bardzo modułowy programista ma możliwość w łatwy sposób rozszerzyć stworzone już funkcjonalności. Z pozytywnych aspektów warto zaznaczyć ze projekt został stworzony przy użyciu technologii Gradle. Jest to nowoczesne rozwiązanie pozwalające budować profesjonalne aplikacje.

Projekt oczywiście nie będzie rozwijany, ale krytycznie patrząc na aplikację można by ją rozwinąć o:

* UX – lepsze wsparcie dla niepełnosprawnych użytkowników, accessibility czy ciekawszy „content” na stworzonych stronach
* Architektonicznie projekt powinien zostać przepisany do Springa który jest nowszą i bardziej stabilną platformą. Dodatkowo przy rozwijaniu aplikacji warto użyć typowych relacyjnych baz danych takich jak postgresql czy mysql. Jeśli natomiast mówimy o funkcjonalności wyszukiwania danych w sposób profesjonalny należy użyć gotowych rozwiązań takich jak np. Elasticsearch.