



Politechnika Wrocławska

Wyniki etapu III: Ocena architektury systemu:
System e-learningowy

Sprawozdanie

Prowadzący:
dr inż. Bogumiła Hnatkowska

Wykonali:
Jakub Staniszewski 266876
Kamil Wojcieszak 264487
Kasjan Kardaś 263505

Wrocław, 27 Styczeń 2026r.

Spis treści

1) Przegląd podejść architektonicznych	3
2) Drzewo użyteczności	3
3) Analiza wybranych scenariuszy	4
4) Punkty wrażliwości i kompromisy	5
5) Ryzyka i nie-ryzyka	5
6) Wnioski	5

1) Przegląd podejść architektonicznych

Architektura ocenianego systemu e-learningowego została zaprojektowana z myślą o spełnieniu kluczowych wymagań jakościowych, takich jak wydajność, skalowalność, bezpieczeństwo, wysoka dostępność oraz zgodność z RODO. System wykorzystuje architekturę warstwową, wdrożoną w środowisku chmurowym.

Podstawowy podział architektury obejmuje:

- **Warstwę prezentacji (Frontend)** - aplikację typu SPA zrealizowaną w technologii React, serwowaną przez Nginx, odpowiedzialną za interakcję z użytkownikiem końcowym.
- **Warstwę logiki biznesowej (Backend)** - aplikację backendową (Python), udostępniającą API obsługujące użytkowników, kursy, testy, postępy oraz mechanizmy realizujące wymagania RODO.
- **Warstwę danych** - relacyjną bazę danych PostgreSQL w konfiguracji klastrowej lub replikacyjnej oraz magazyn obiektowy Amazon S3, przeznaczony do przechowywania plików multimedialnych.

Komunikacja pomiędzy warstwami realizowana jest wyłącznie poprzez zdefiniowane interfejsy API, co zapewnia luźne powiązania pomiędzy komponentami oraz umożliwia ich niezależny rozwój i skalowanie. System zostanie wdrożony w architekturze wieloinstancyjnej, z wykorzystaniem mechanizmów autoskalowania i load balancingu, eliminujących pojedyncze punkty awarii.

W zakresie bezpieczeństwa zastosowano szyfrowanie całej komunikacji z użyciem protokołu TLS oraz bezpieczne przechowywanie haseł użytkowników w postaci skrótów bcrypt z losową solą. Wymagania RODO realizowane są poprzez wydzielony moduł logiczny backendu, odpowiedzialny za kontrolowane przetwarzanie danych osobowych, ich eksport oraz trwałe usuwanie.

2) Drzewo użyteczności

Atrybut jakości	Udoskonalenie atrybutu	Scenariusze
Wydajność i skalowalność	Czas odpowiedzi na zapytanie	Utrzymanie czasu odpowiedzi API na poziomie średnio 300ms (95 percentyl)
	Liczba użytkowników	Wsparcie dla co najmniej 1 000 równoczesnych użytkowników bez spadku wydajności
Bezpieczeństwo	Ochrona danych	Wartości haseł wszystkich użytkowników nie mogą być możliwe do przeczytania
	Ochrona komunikacji	Brak możliwości przechwycenia przez użytkownika postronnego komunikacji między klientem a serwerem
Dostępność	Czas dostępności	Zapewnienie dostępności do aplikacji przez co najmniej 99% czasu
	Skalowalność	System musi obsłużyć wzmożony ruch użytkowników w okresach szczytowych
RODO	Pozyskiwanie danych	Mechanizm umożliwiający pobranie danych osobowych (JSON/CSV)
	Pozbywanie się danych	Możliwość całkowitego usunięcia konta wraz z danymi w ciągu maksymalnie 30 dni

Tabela 1: Drzewo użyteczności - atrybuty jakości i scenariusze

3) Analiza wybranych scenariuszy

Scenariusz: A1	System musi obsłużyć wzmożony ruch użytkowników w okresach szczytowych			
Atrybuty	Wydajność, Skalowalność			
Środowisko	Normalny tryb pracy			
Bodziec	Jednoczesne korzystanie z systemu przez ponad 1000 użytkowników			
Odpowiedź	System utrzymuje średni czas odpowiedzi API na poziomie 300 ms (95 percentyl) bez spadku dostępności usług.			
Decyzje architektoniczne	Wrażliwość	Kompromis	Ryzyko	Brak ryzyka
Auto Scaling Group – Backend	S1	T1		N1
Auto Scaling Group – Frontend	S2	T2	R1	
Analiza	Zastosowanie autoskalowania backendu i frontendu umożliwia dynamiczne dostosowanie zasobów do aktualnego obciążenia systemu. Wrażliwość S1 i S2 dotyczy poprawnej konfiguracji progów skalowania. Ryzyko R1 występuje w przypadku zbyt wolnej reakcji autoskalera, co może prowadzić do chwilowych opóźnień odpowiedzi systemu.			

Scenariusz: A1	System musi obsłużyć wzmożony ruch użytkowników w okresach szczytowych			
Atrybuty	Wydajność, Skalowalność			
Środowisko	Normalny tryb pracy			
Bodziec	Jednoczesne korzystanie z systemu przez ponad 1000 użytkowników			
Odpowiedź	System utrzymuje średni czas odpowiedzi API na poziomie 300 ms (95 percentyl) bez spadku dostępności usług.			
Decyzje architektoniczne	Wrażliwość	Kompromis	Ryzyko	Brak ryzyka
Auto Scaling Group – Backend	S1	T1		N1
Auto Scaling Group – Frontend	S2	T2	R1	
Analiza	Zastosowanie autoskalowania backendu i frontendu umożliwia dynamiczne dostosowanie zasobów do aktualnego obciążenia systemu. Wrażliwość S1 i S2 dotyczy poprawnej konfiguracji progów skalowania. Ryzyko R1 występuje w przypadku zbyt wolnej reakcji autoskalera, co może prowadzić do chwilowych opóźnień odpowiedzi systemu.			

4) Punkty wrażliwości i kompromisy

S1:

Mechanizm autoskalowania backendu jest wrażliwy na nieprawidłowo dobrane progi obciążenia. Zbyt późne uruchamianie nowych instancji może prowadzić do chwilowego spadku wydajności systemu.

S2:

Autoskalowanie warstwy frontendowej wpływa na zdolność obsługi dużej liczby jednoczesnych połączeń HTTP. Wrażliwość dotyczy synchronizacji skalowania frontendu z backendem.

T1:

Autoskalowanie backendu umożliwia obsługę zwiększonego obciążenia kosztem większej złożoności infrastruktury i konieczności monitorowania.

T2:

Skalowanie frontendu poprawia dostępność aplikacji, ale generuje dodatkowe koszty infrastrukturalne.

5) Ryzyka i nie-ryzyka

R1:

Nieprawidłowa konfiguracja mechanizmów autoskalowania może prowadzić do okresowych spadków wydajności przy gwałtownych wzrostach ruchu użytkowników.

N1:

Zastosowanie autoskalowania backendu umożliwia spełnienie wymagań wydajnościowych przy dużej liczbie użytkowników.

6) Wnioski