

# Ocena Architektury

## Zespół oceniający:

Tomasz Mosur 228068  
Paweł Głuszcak 228109  
Marcin Szeptun 228043

## Oceniane:

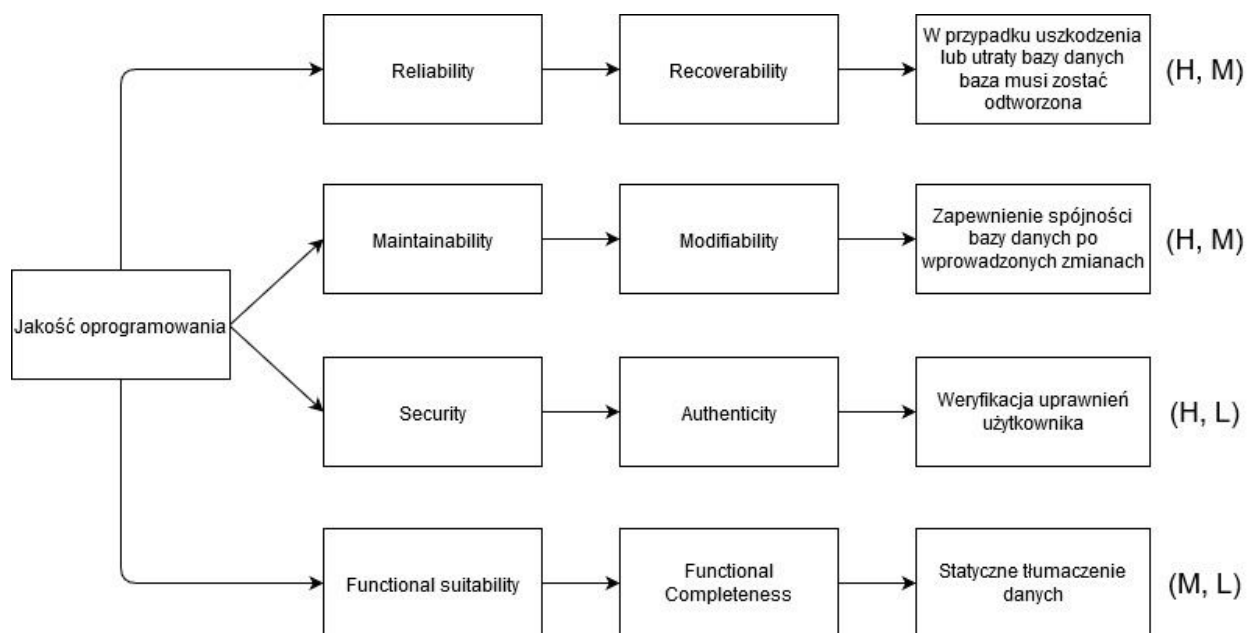
*Tworzenie i modyfikowanie programów kształcenia*

Rafał Bukowski 226156  
Ania Pijanowska 228055  
Mateusz Najda 225930

## Przegląd podejść architektonicznych

W projekcie zastosowano architekturę klient serwer. Zastosowano uwierzytelnianie poprzez zewnętrzny system. Generowanie skryptów bazy z modelu ERD odbędzie się z wykorzystaniem dodatkowej aplikacji. System w celu pobrania danych integruje się z systemami zewnętrznymi. System oraz baza danych znajdują się jednym serwerze. Architektura zakłada wsparcie wielu języków dla statycznych elementów interfejsu.

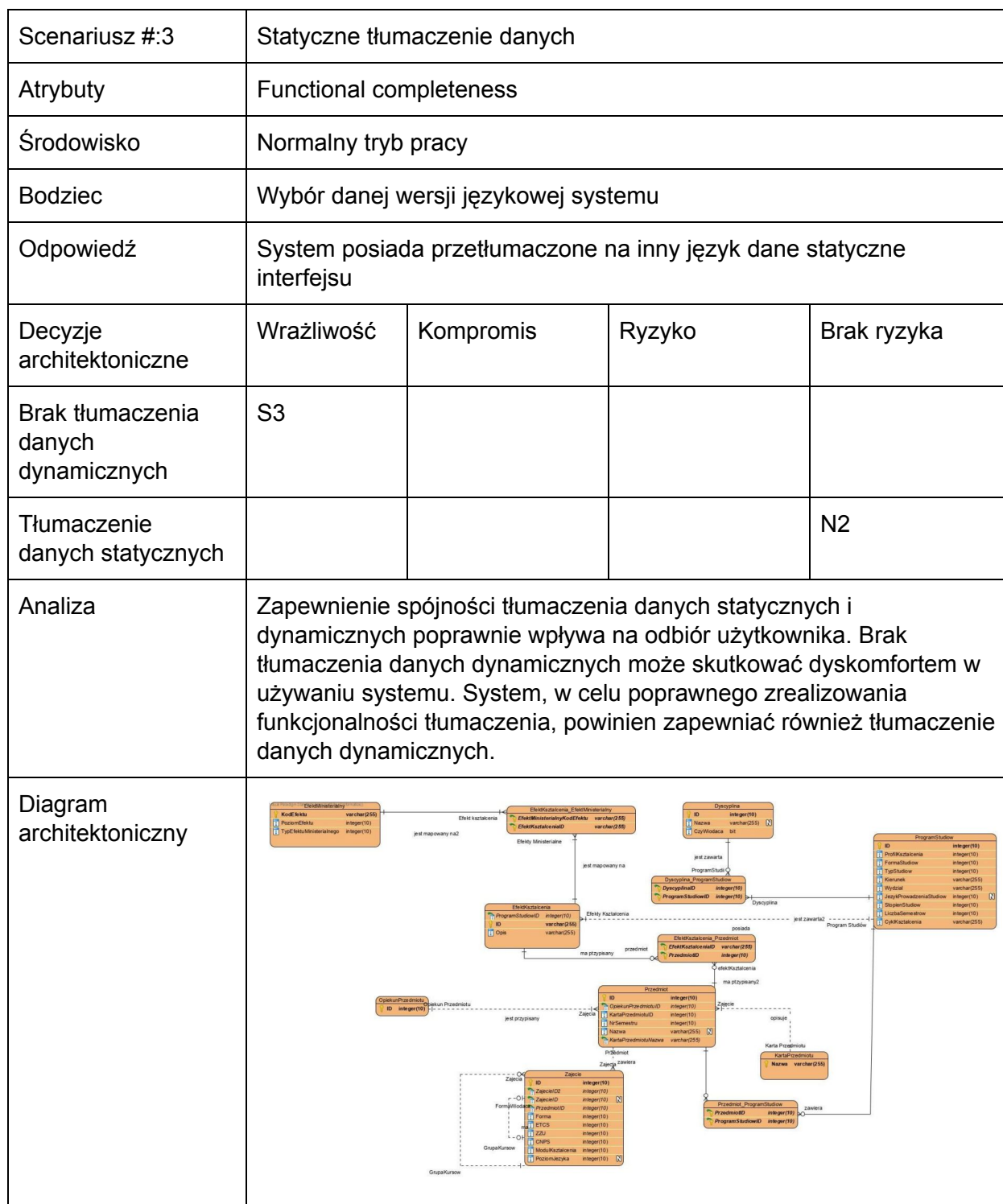
## Drzewo użyteczności

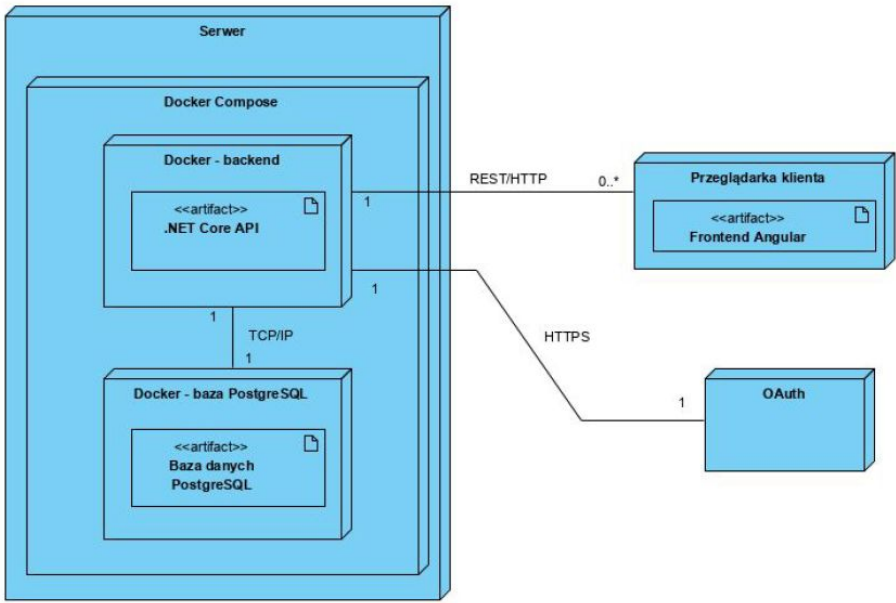


## Analiza wybranych scenariuszy

Scenariusz #:1	Zapewnienie spójności bazy danych po wprowadzonych zmianach			
Atrybuty	Maintainability			
Środowisko	Środowisko deweloperskie			
Bodziec	Wprowadzona zmiana w modelu bazy danych			
Odpowiedź	Zaktualizowanie schematu bazy danych w środowiskach deweloperskich i produkcyjnych oraz migracja danych			
Decyzje architektoniczne	Wrażliwość	Kompromis	Ryzyko	Brak ryzyka
Generowanie nowego schematu bazy danych z użyciem Visual Paradigm			R1	
Zapewnienie spójności danych	S1		R2	
Zapewnienie spójności schematu we wszystkich środowiskach	S2			
Analiza	Brak informacji o mechanizmach aktualizacji bazy danych i migracji danych. Po aktualizacji bazy danych i wygenerowaniu skryptu tworzącego bazę w dodatkowej aplikacji, aktualnie znajdujące się w niej dane zostaną utracone. Po zaktualizowaniu bazy danych, nie ma mechanizmu zapewnienia spójności schematu we wszystkich środowiskach.			

Scenariusz #:2	Weryfikacja uprawnień użytkownika			
Atrybuty	Authenticity			
Środowisko	Normalny tryb pracy			
Bodziec	Logowanie do systemu			
Odpowiedź	Uprawnienia użytkownika definiują dostępne funkcjonalności			
Decyzje architektoniczne	Wrażliwość	Kompromis	Ryzyko	Brak ryzyka
Autoryzacja z OAuth 2.0		T1		N1
Analiza	<p>W systemie funkcjonalności są uzależnione od uprawnień użytkownika. Uprawnienia te są ustalane na podstawie uwierzytelnienia użytkownika za pomocą zewnętrznego systemu. System główny jest uzależniony od poprawności działania systemu zewnętrznego, co w większości przypadków jest obciążeniem systemu głównego, jednak w przypadku awarii systemu zewnętrznego system główny również przestaje działać.</p>			



Scenariusz #:4	W przypadku uszkodzenia lub utraty bazy danych baza musi zostać odtworzona			
Atrybuty	Reliability			
Środowisko	Awaria systemu			
Bodziec	Baza danych zostaje utracona			
Odpowiedź	Przywrócenie normalnego działania bazy danych.			
Decyzje architektoniczne	Wrażliwość	Kompromis	Ryzyko	Brak ryzyka
Jeden serwer dla systemu i bazy danych		T2	R3	
Tworzenie kopii zapasowych bazy danych				N3
Analiza	W zaproponowanym rozwiązaniu utrzymywany zostaje tylko jeden serwer co znacząco obniża koszty utrzymania systemu. W przypadku awarii mechanicznych serwera znacząco utrudnione jest odzyskanie danych oraz sprawności systemu.			
Diagram architektoniczny	 <pre> graph LR     subgraph Server         subgraph DockerCompose [Docker Compose]             subgraph DockerBackend [Docker - backend]                 API[&lt;&lt;artifact&gt;&gt; .NET Core API]             end             subgraph DockerDB [Docker - baza PostgreSQL]                 DB[&lt;&lt;artifact&gt;&gt; Baza danych PostgreSQL]             end             API -- "1 TCP/IP 1" --- DB         end     end     API -- "1 REST/HTTP 0..*" --- FE[Przeglądarka klienta&lt;br/&gt;&lt;&lt;artifact&gt;&gt; Frontend Angular]     API -- "1 HTTPS 1" --- OAuth[OAuth] </pre>			

## Punkty wrażliwości i kompromisy

S1. Przy zmianie schematu bazy danych, dane mogą być niepoprawnie przeniesione z poprzedniego schematu.

S2. Baza danych może być niespójna pomiędzy różnymi wersjami deweloperskimi i produkcyjnymi.

S3. Brak tłumaczenia danych dynamicznych przy przetłumaczonych danych statycznych może prowadzić do zaburzenia doświadczeń użytkownika korzystającego z systemu.

T1. Autoryzacja z wykorzystaniem zewnętrznego serwera OAuth 2.0 ma pozytywny wpływ na bezpieczeństwo. Przy wykorzystywaniu zewnętrznego serwera nie ma pełnej kontroli nad jego pracą.

T2. Przechowywanie systemu i bazy danych na jednym serwerze zmniejszy ogólny koszt utrzymania systemu.

## Ryzyka i nie-ryzyka

R1. Generowanie nowego schematu bazy danych, przy każdej zmianie modelu, z użyciem oddzielnego narzędzia może prowadzić do nieoczekiwanych zmian tego schematu.

R2. Zmiana schematu bazy danych może prowadzić do zmian, które uniemożliwiają spójne przeniesienie danych z poprzedniego schematu. W takim przypadku dane zostaną utracone.

R3. Utrzymywanie całego systemu na jednym serwerze prowadzi do zwiększenia ryzyka utraty danych oraz działania całego systemu w przypadku uszkodzeń mechanicznych.

N1. System OAuth 2.0 jest renomowanym i polecanym systemem do autoryzacji toteż można założyć bezpieczne wykorzystywanie go.

N2. Tłumaczenie danych statycznych, ze względu na ich pewną obecność w systemie, jest poprawnym rozwiązaniem.

N3. Tworzenie kopii zapasowych bazy danych jest decyzją poprawną, zgodną z regułami sztuki.

## Wnioski

Najbardziej prawdopodobne ryzyka dotyczą bazy danych, m.in. jej modyfikacji, zapewnienia spójności oraz zapewnienia kopii zapasowej danych. W proponowanym rozwiązaniu, w przypadku awarii dysku serwera, oprócz wystąpienia problemu z działaniem systemu, może dojść do utraty wszystkich danych z bazy danych. Strategia generowania i aktualizowania schematu bazy danych zaproponowana w architekturze może prowadzić do niespójności w systemie oraz grozi utratą danych podczas migracji.

Zarówno wykorzystanie systemu OAuth 2.0, tłumaczenie danych statycznych oraz tworzenie kopii zapasowych bazy danych są decyzjami poprawnymi, zgodnymi ze sztuką i pozytywnie wpływają na realizację funkcjonalności systemu.