

Projekt zespołowy – etap 3

SYSTEM WSPOMAGAJĄCY DZIAŁANIE FIRMY SPRZĄTAJĄCEJ

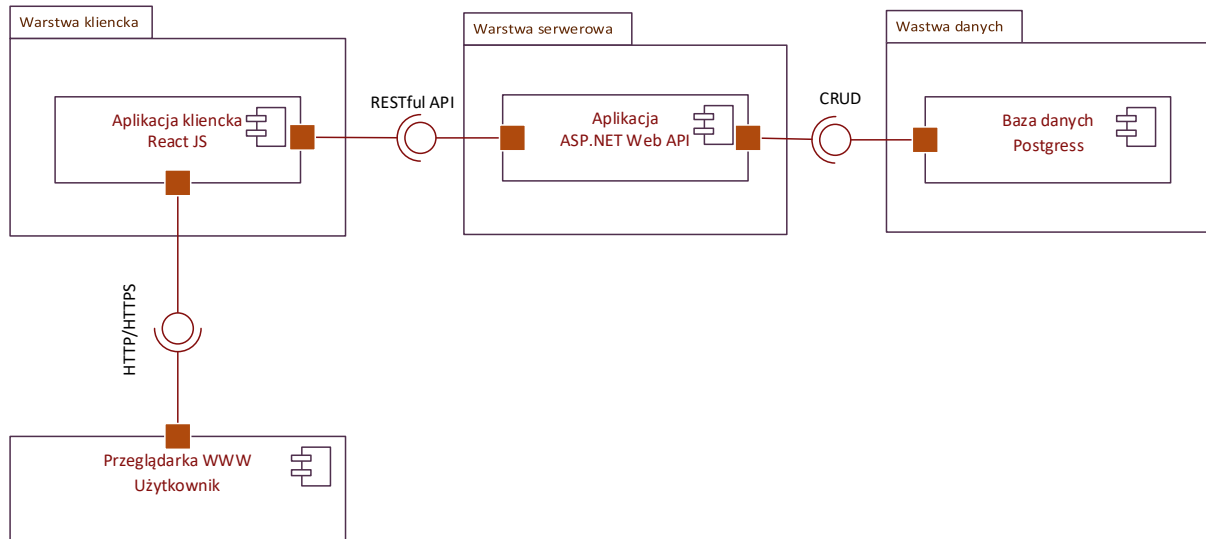
MARCIN OSIADACZ, KACPER KOWALSKI, TOBIASZ PIWOWARCZYK, WIKTOR DMITRUK

Spis treści

1. Projekt architektury systemu	2
1.1. Warstwa danych	2
1.2. Warstwa serwerowa	2
1.3. Warstwa kliencka	3
2. Projekt bazy danych	5
2.1. Diagram ERD	6
2.2. Opis poszczególnych encji oraz relacji	7
2.2.1 ApplicationUser	7
2.2.2 Order	8
2.2.3 OrderToServicePrice	8
2.2.4 OrderStatus	9
2.2.5 ServicePrice	9
3. Projekt algorytmów	10
3.1 Niezalogowany użytkownik	10
3.2 Klient	11
3.3 Pracownik	12
3.4 Administrator	13
4. Projekt interfejsu użytkownika	15
5. Projekt testów	20
5.1. Testy funkcjonalne	20
5.2. Testy zgodności	23
5.3. Testy wydajnościowe	23

1. Projekt architektury systemu

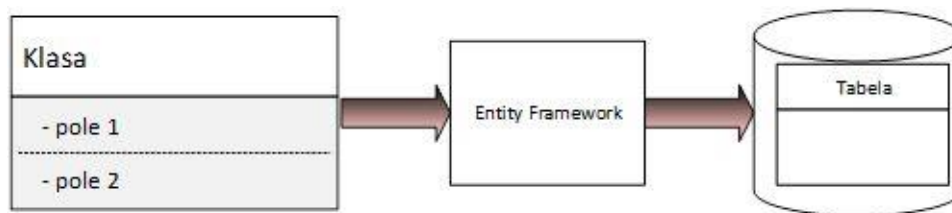
Budowana aplikacja wspierająca zarządzanie firmą sprzętającą charakteryzować się będzie strukturą trójwarstwową, na którą składać się będą warstwa danych, warstwa serwerowa oraz warstwa kliencka. Dostęp do aplikacji realizowany będzie poprzez przeglądarkę internetową wykorzystującą protokół http/HTTPS. Struktura ta została przedstawiona na Rysunku 1.



Rysunek 1 Schemat architektury systemu.

1.1. Warstwa danych

Narzędziem wykorzystanym do składowania danych będzie relacyjna baza danych PostgreSQL w wersji 14.1. Baza ta będzie udostępniać standardowy interfejs do tworzenia, edycji, dostarczania i usuwania rekordów (CRUD). Struktura bazy danych zostanie wygenerowana przy pomocy podejścia Code First (Rysunek 2) z wykorzystaniem Entity Framework w wersji szóstej. Framework ten zostanie również wykorzystanych do zasilenia bazy danych danymi słownikowymi.

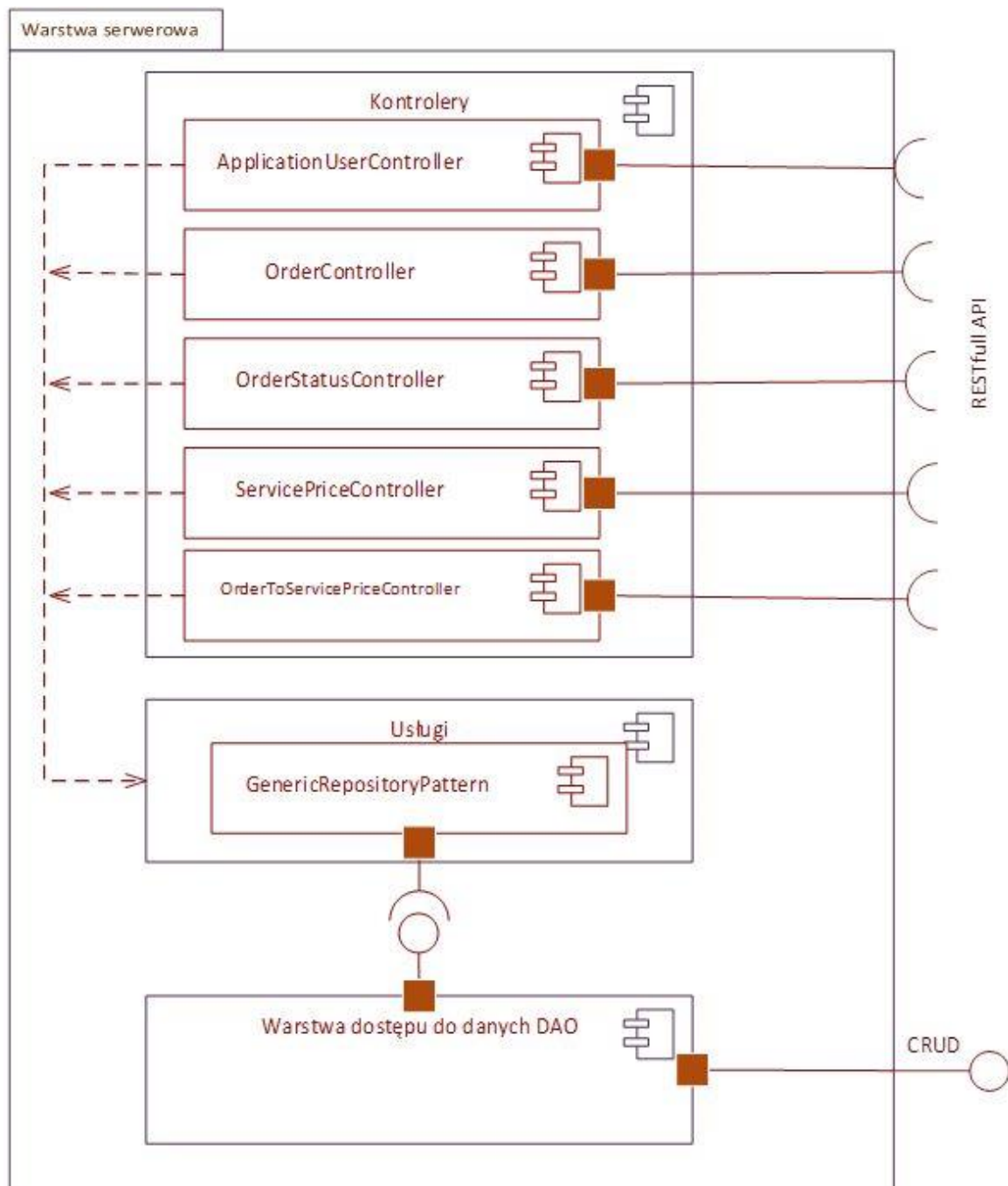


Rysunek 2 Schemat podejścia *Code First*.

1.2. Warstwa serwerowa

Rolą warstwy serwerowej jest przyjmowanie poleceń z aplikacji frontendowej w postaci poleceń RESTful API i przekazywanie oraz pobieranie informacji do i z bazy danych. W warstwie tej możemy wyróżnić kontrolery, których rolą jest grupowanie i definiowanie żądań do poszczególnych encji. Żądania te są mapowane na akcje wykorzystując mechanizm routingu. W systemie wystąpi pięć kontrolerów, które będą obsługiwać żądania http typu GET, POST, PUT i DELETE. Każdy z kontrolerów implementuje wzorzec generycznego repozytorium zapewniających podstawową implementację wcześniej wskazanych danych poprzez wstrzykiwanie zależności (DI). Generyczne repozytorium zapewnia asynchroniczny dostęp do kontekstu bazy danych i przesyła zapytania do bazy danych.

Na tym etapie walidacja danych ograniczona jest do sprawdzania typów zmiennych przekazywanych, natomiast walidacja biznesowa realizowana jest na poziomie aplikacji klienckiej. Dostęp do bazy danych zapewniany jest poprzez mechanizm DbContext frameworka Entity.



Rysunek 3 Schemat warstwy serwerowej.

1.3. Warstwa kliencka

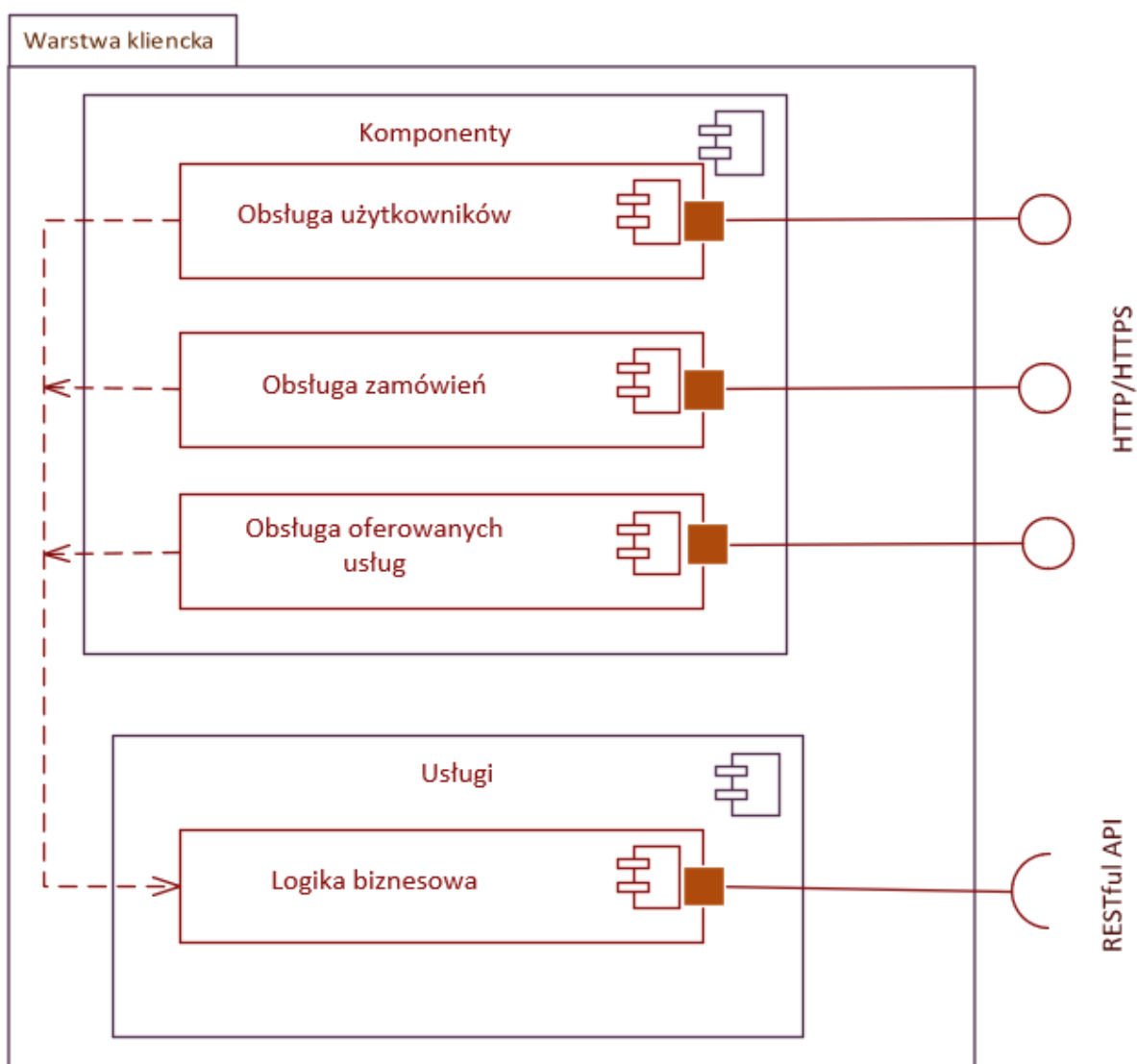
Aplikacja kliencka zostanie zrealizowana z wykorzystaniem frameworka React JS wykorzystując podejście zorientowane na komponenty. Podejście to ma za zadanie przesunięcie nacisku na wykorzystanie w procesie projektowania systemu na dostępne, gotowe, wykorzystywane wielokrotnie moduły, który jedynie część zostanie wytworzona na potrzeby opracowywanej aplikacji. Zaletą takiego postępowania jest zwiększenie niezawodności aplikacji, gdyż użyte gotowe elementy były wielokrotnie

sprawdzone wcześniej. Nie bez znaczenia jest zmniejszenie czasu koniecznego na implementację, ze względu na mniejszą konieczność wytworzenia specyficznego, dedykowanego kodu. Komunikacja pomiędzy poszczególnymi komponentami odbywa się za pośrednictwem dobrze udokumentowanych interfejsów co powoduje, że mniejsze jest ryzyko ich kolizji.

Warstwa kliencka będzie miała za zadanie dostarczanie użytkownikowi końcowemu interfejsu webowego umożliwiającego interakcję z trzema grupami aktywności w systemie:

- obsługą użytkowników,
- obsługą zamówień,
- obsługą oferowanych usług.

Aktywności te będą zależeć od warstwy logiki biznesowej, której zadaniem będzie walidacją danych wprowadzanych przez użytkownika oraz komunikacją z warstwą serwerową poprzez RESTful API.



Rysunek 4 Schemat warstwy klienckiej.

2. Projekt bazy danych

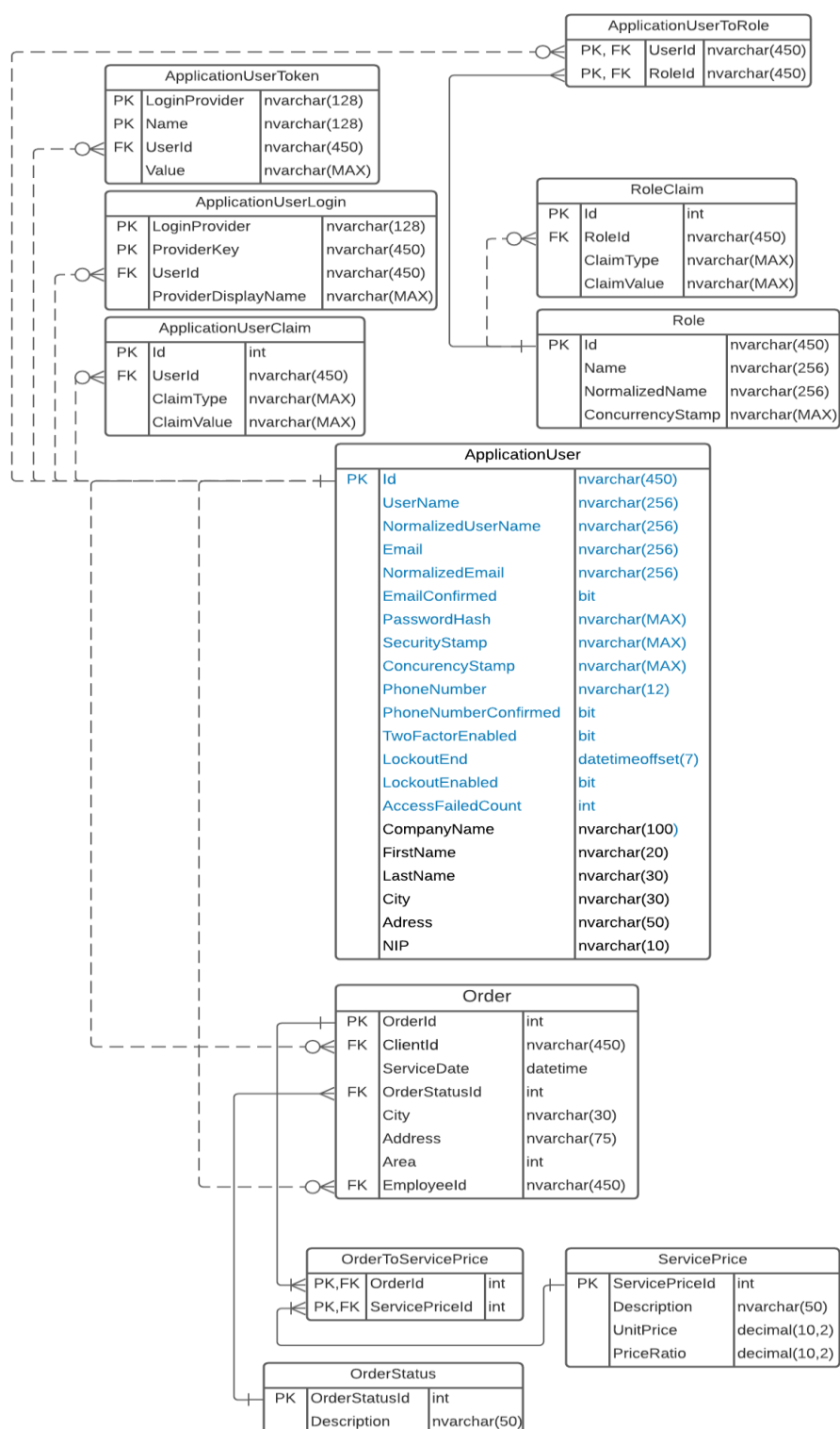
Na potrzeby realizacji projektu informacje przechowywane będą w jedenastu tabelach bazy danych. W pierwszych siedmiu tabelach będą znajdować się dane systemowe konieczne do zarządzania kontami użytkowników - w szczególności ich hasłami, rolami, danymi profilu, poświadczeniami oraz posiadanymi tokenami. Funkcjonalność ta zrealizowana zostanie poprzez implementację interfejsu API z biblioteki Identity frameworka ASP.NET Core. Dane główne, generowane w trakcie funkcjonowania systemu zbierane będą w pięciu tabelach, z których jedna (ApplicationUser) będzie wspólna zarówno dla klientów jak i użytkowników serwisu. W tej encji pola oznaczone kolorem niebieskim są polami dziedziczonymi encji ASPNetUsers.

Na model danych składają się następujące encje:

- a. **ApplicationUserToken, ApplicationUserLogin, ApplicationUserClaim, ApplicationUserToRole, RoleClaim, Role** - dane systemowe, niezbędne do procesu rejestracji i logowania użytkowników;
- b. **ApplicationUser** - klienci oraz użytkownicy systemu;
- c. **Order** - zamówienia klientów;
- d. **OrderStatus** - słownik statusu zamówień;
- e. **ServicePrice** - słownik cennika usług;
- f. **OrderToServicePrice** - usługi z cennika wybrane przez klienta do realizacji zamówienia.

2.1. Diagram ERD

Graficzna reprezentacja zaimplementowanego modelu danych przedstawia poniższy diagram ERD (*Entity Relationship Diagram*) wykonany w notacji Martina, zwaną również notacją “kurzych łapek”.



Rysunek 5 Diagram ERD

2.2. Opis poszczególnych encji oraz relacji

Encje pochodzące z pakietu ASP.NET Core Identity wymienione w punkcie 3 a są niezmodyfikowane i charakteryzują się standardowymi, predefiniowanymi powiązaniem. Jedyną zmianą jest dostosowanie nazw do pozostałych encji modelu. Reasumując kwestie związane z zarządzaniem użytkownikiem należy wspomnieć, że encję ApplicationUser z encjami ApplicationUserClaim, ApplicationUserLogin i ApplicationUserToken wiąże opcjonalna relacja jeden do wielu (1:N). Ta sama encja tj. ApplicationUser wiązana jest z encją Role relacją wiele do wielu (N:N), gdzie encją pośredniczącą jest encja ApplicationUserToRole. Ponadto, każda z encji Role może charakteryzować się wieloma skojarzonymi encjami RoleClaim (1:N).

Istotniejszymi z punktu widzenia są pozostałe encje gromadzące dane użytkowników. W poniższych tabelach znajdują się definicje tych encji.

2.2.1 ApplicationUser

Encja z danymi klientów i pracowników.

Klucz podstawowy: Id

Klucze obce: brak

Relacje: opisane powyżej i poniżej

Pola	Typ danych	Czy puste	Opis
(..) pola odziedziczone z encji AspNetUser			
CompanyName	nvarchar(450)	TAK	Nazwa firmy, jeżeli jest to użytkownik firmowy
FirstName	nvarchar(20)	TAK	Imię, jeżeli jest to użytkownik prywatny
LastName	nvarchar(30)	TAK	Nazwisko, jeżeli jest to użytkownik prywatny
City	nvarchar(30)	TAK	Miasto pochodzenia
Address	nvarchar(50)	TAK	Adres użytkownika
NIP	nvarchar(10)	TAK	NIP firmy, jeżeli jest to użytkownik firmowy

2.2.2 Order

Encja z zamówieniami klientów

Klucz podstawowy: OrderId

Klucze obce: ClientId->ApplicationUser(Id),
EmployeeId->Applicationuser(Id)
OrderStatusId->OrderStatus(Id)

Relacje:

- **ApplicationUser (Id) – Order (ClientId) , ApplicationUser(Id) – Order (EmployeeId)** obie typu jeden do wielu (1:N), obie opcjonalne, gdyż nie każdy użytkownik musi mieć zleconą usługę sprzątnia i nie każdy pracownik musi mieć przypisane zlecenie sprzątnia.
- **Order(OrderStatusId) - OrderStatus(Id)** typu 1:N wiążąca zlecenie z jej statusem. Relacja ta jest jednoznaczna tzn, każde zlecenie musi mieć zdefiniowany status.

Pola	Typ danych	Czy puste	Opis
OrderId	int	NIE	Nazwa firmy, jeżeli jest to użytkownik firmowy
ClientId	int	NIE	Imię, jeżeli jest to użytkownik prywatny
ServiceDate	datetime	NIE	Nazwisko, jeżeli jest to użytkownik prywatny
OrderStatusId	int	NIE	Status zlecenia
City	nvarchar(30)	NIE	Miasto, wykonania usługi
Address	nvarchar(50)	NIE	Adres, lokalu do wykonania usługi
Area	int	NIE	Powierzchnia lokalu przeznaczonego do usługi
EmployeeId	int	TAK	Identyfikator pracownika przypisanego do zlecenia

2.2.3 OrderToServicePrice

Encja pośrednicząca pomiędzy zamówieniem a przedmiotem zamówienia

Klucz podstawowy: sprzężony z pól: OrderId i ServicePricId

Klucze obce: OrderId->Order (Id),
ServicePricId->ServicePrice(Id)

Relacje:

- **Order (Id) – OrderToServicePrice (OrderId) , ServicePrice(Id) – OrderToServicePrice (ServicePricId)** obie typu jeden do wielu (1:N), jednoznaczne, gdyż nie ma zlecenia bez wskazania jego zakresu. Encja ta jest encją pośredniczącą pomiędzy encjami Order i ServicePrice, które łączy relacja wiele do wielu (N:N).

Pola	Typ danych	Czy puste	Opis
OrderId	int	NIE	Identyfikator zlecenia
ServicePriceId	int	NIE	Identyfikator usługi słownikowej

2.2.4 OrderStatus

Encja słownikowa z możliwymi statusami zamówienia

Klucz podstawowy: OrderStatusId

Klucze obce: brak

Relacje: powyżej

Pola	Typ danych	Czy puste	Opis
OrderId	int	NIE	Identyfikator zlecenia
ServicePriceId	int	NIE	Identyfikator usługi słownikowej

2.2.5 ServicePrice

Encja słownikowa z cennikiem usług

Klucz podstawowy: ServicePriceId

Klucze obce: brak

Relacje: powyżej

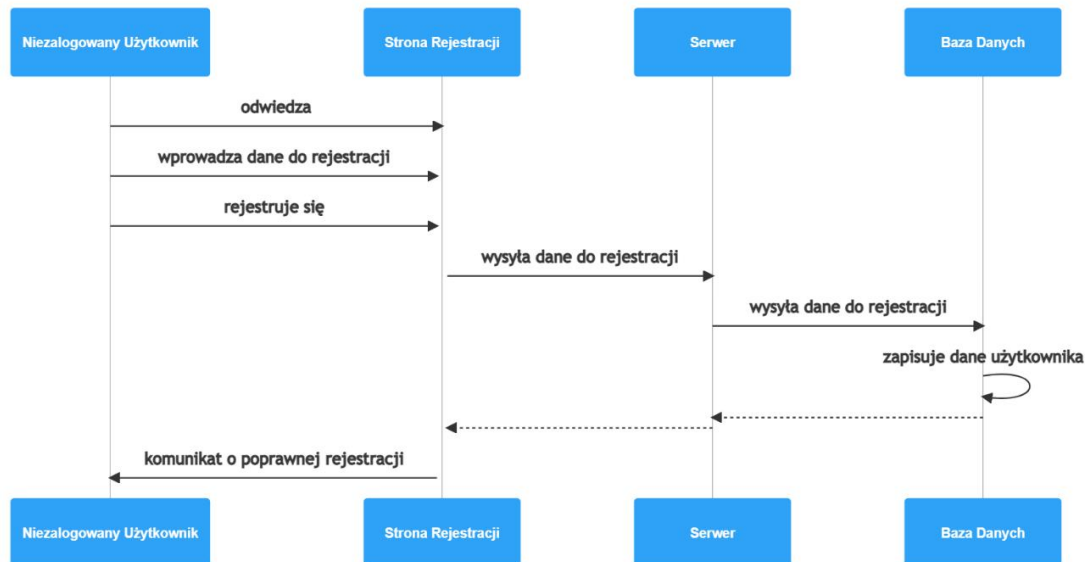
Pola	Typ danych	Czy puste	Opis
ServicePriceId	int	NIE	Identyfikator pozycji słownika
DescriptionPrice	nvarchar(50)	NIE	Nazwa usługi
UnitPrice	decimal(10,2)	NIE	Koszt jednostkowy usługi
PriceRatio	decimal(10,2)	NIE	Współczynnik wiążący cenę usługi z metrażem lokalu do sprzątnia

3. Projekt algorytmów

3.1 Niezalogowany użytkownik

Niezalogowany użytkownik ma możliwość:

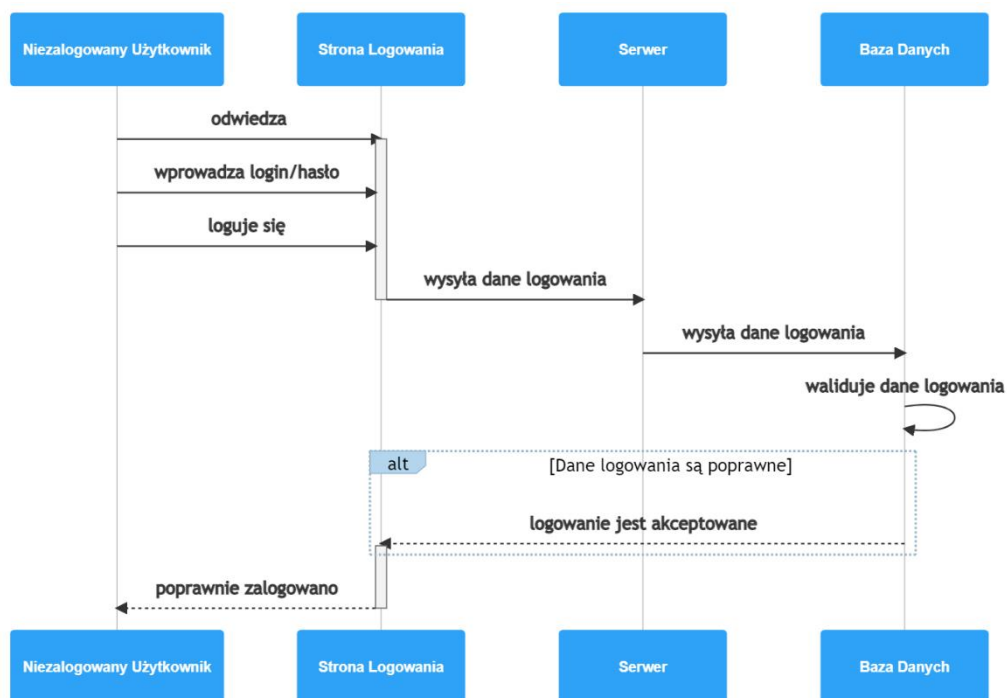
1. Rejestracji konta



Rysunek 6 Rejestracja Konta

Niezalogowany użytkownik ma możliwość odwiedzenia Strony Rejestracji. Po wypełnieniu formularza rejestracyjnego i kliknięciu w przycisk 'Wyślij' dane wysyłane są na Serwer a następnie zapisywane w Bazie danych. Następnie, wyświetlony zostaje komunikat o poprawnej rejestracji nowego użytkownika.

2. Logowania na konto



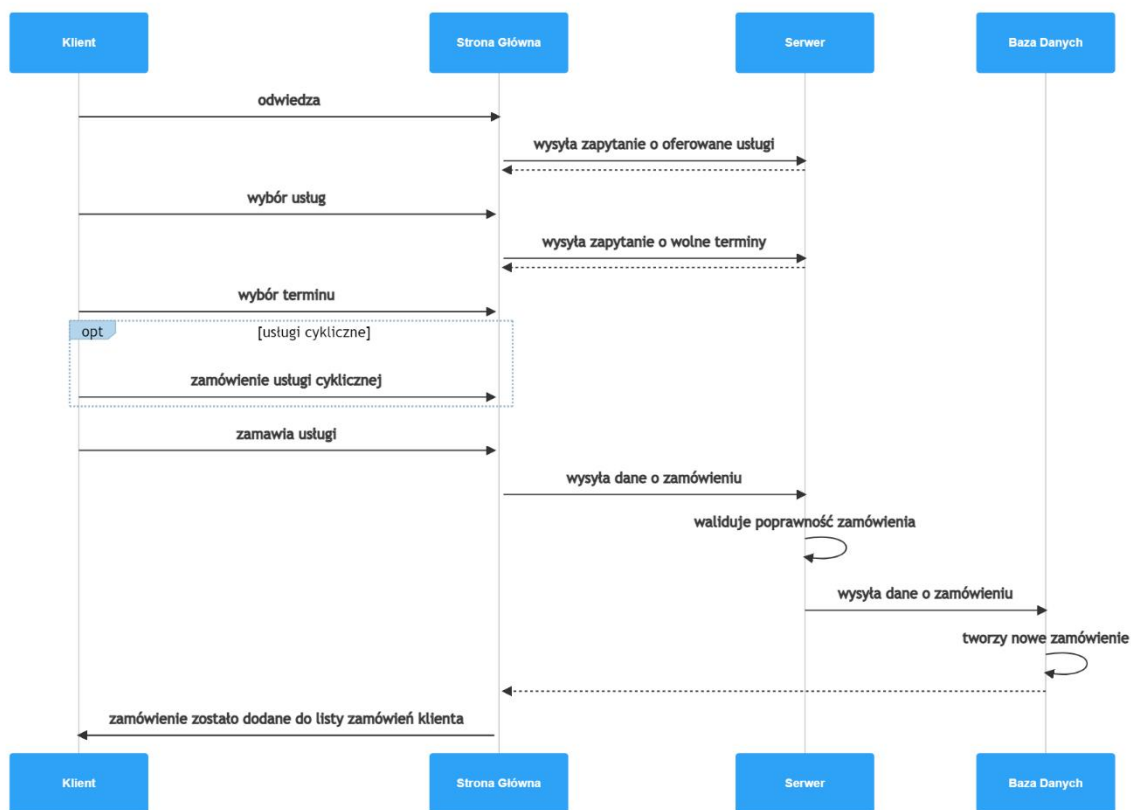
Rysunek 7 Logowanie

Niezałogowany użytkownik ma również możliwość logowania. Po wejściu na Stronę Logowania i wprowadzeniu loginu i hasła, może kliknąć w przycisk 'Zaloguj'. Dane logowania wysyłane są na Serwer a następnie do Bazy danych. Po ich walidacji użytkownik zostaje poprawnie zalogowany.

3.2 Klient

Po zalogowaniu użytkownik może dodatkowo:

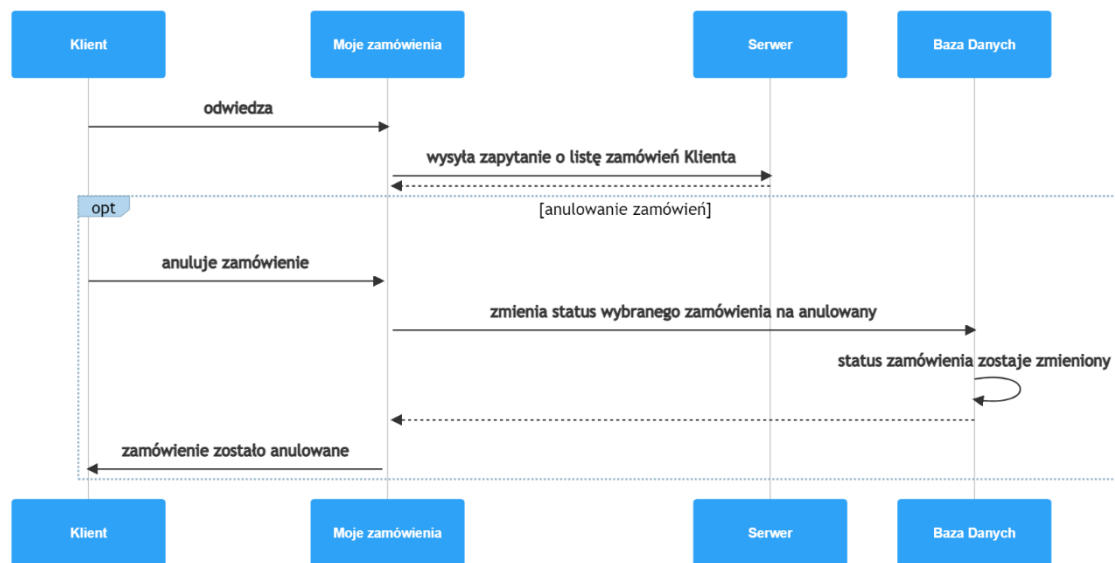
1. Złożyć zamówienie na sprzętanie jednorazowe lub cykliczne



Rysunek 8 Składanie zamówień

Zalogowany użytkownik może odwiedzić Stronę Główną aplikacji. Gdy z Serwera zostaną pobrane oferowane usługi, może dokonać ich wyboru. Następnie dokonuje wyboru terminu, z opcją wyboru usługi cyklicznej. Po kliknięciu w przycisk 'Zamów' dane o nowym zamówieniu wysyłane są na Serwer i walidowane. Po ukończeniu walidacji, w bazie danych tworzone jest nowe zamówienie a Klientowi przedstawiony jest komunikat zwrotny o poprawnie utworzonym zamówieniu.

2. Przeglądać i anulować złożone przez siebie zamówienia

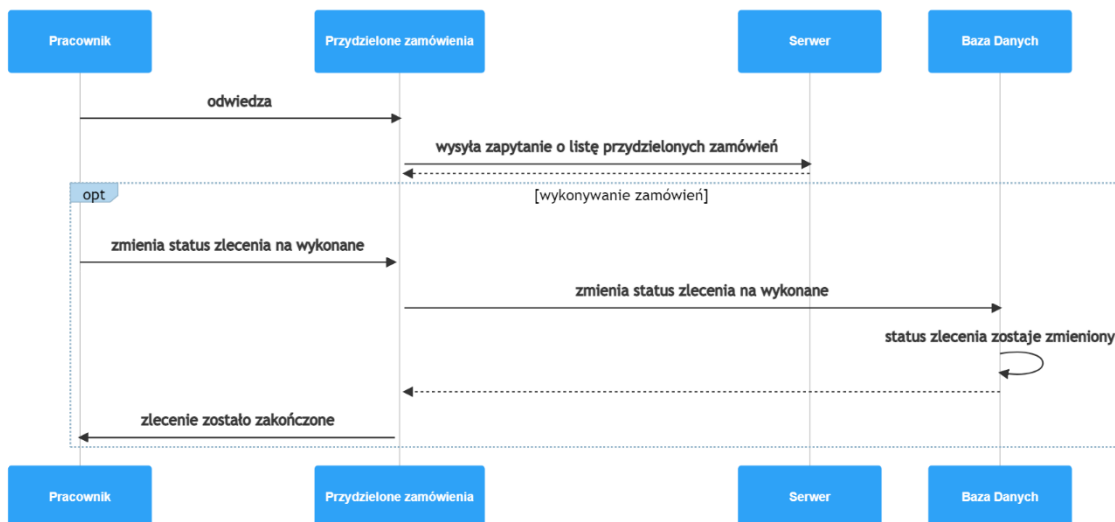


Rysunek 9 Zarządzanie zamówieniami

Ponadto, Klient ma możliwość zarządzania złożonymi przez siebie zamówieniami. Po odwiedzeniu strony Moje zamówienia, Klient widzi listę wszystkich dokonanych przez siebie zamówień. Ma on też możliwość anulowania stworzonego zamówienia. W tym przypadku status zamówienia w Bazie danych ulega zmianie, a Użytkownik otrzymuje komunikat o poprawnie anulowanym zamówieniu.

3.3 Pracownik

Pracownik może przeglądać przydzielone do siebie zamówienia jak i oznaczyć je jako wykonane.



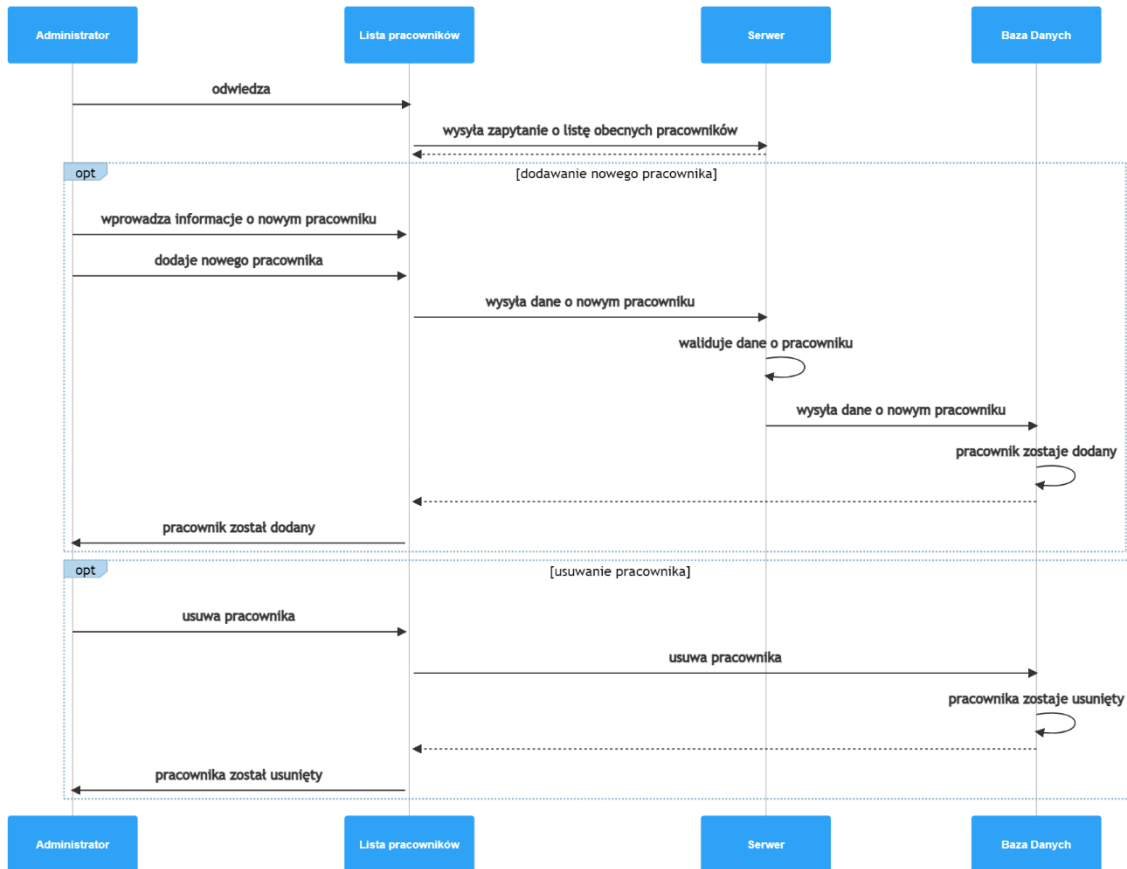
Rysunek 10 Zarządzanie przydzielonymi zamówieniami

Mechanizm działania tych operacji jest praktycznie identyczny do anulowania zamówień przez Klienta.

3.4 Administrator

Administrator ma możliwość:

1. Przeglądania bazy pracowników jak i dodawania i usuwania kont pracowników

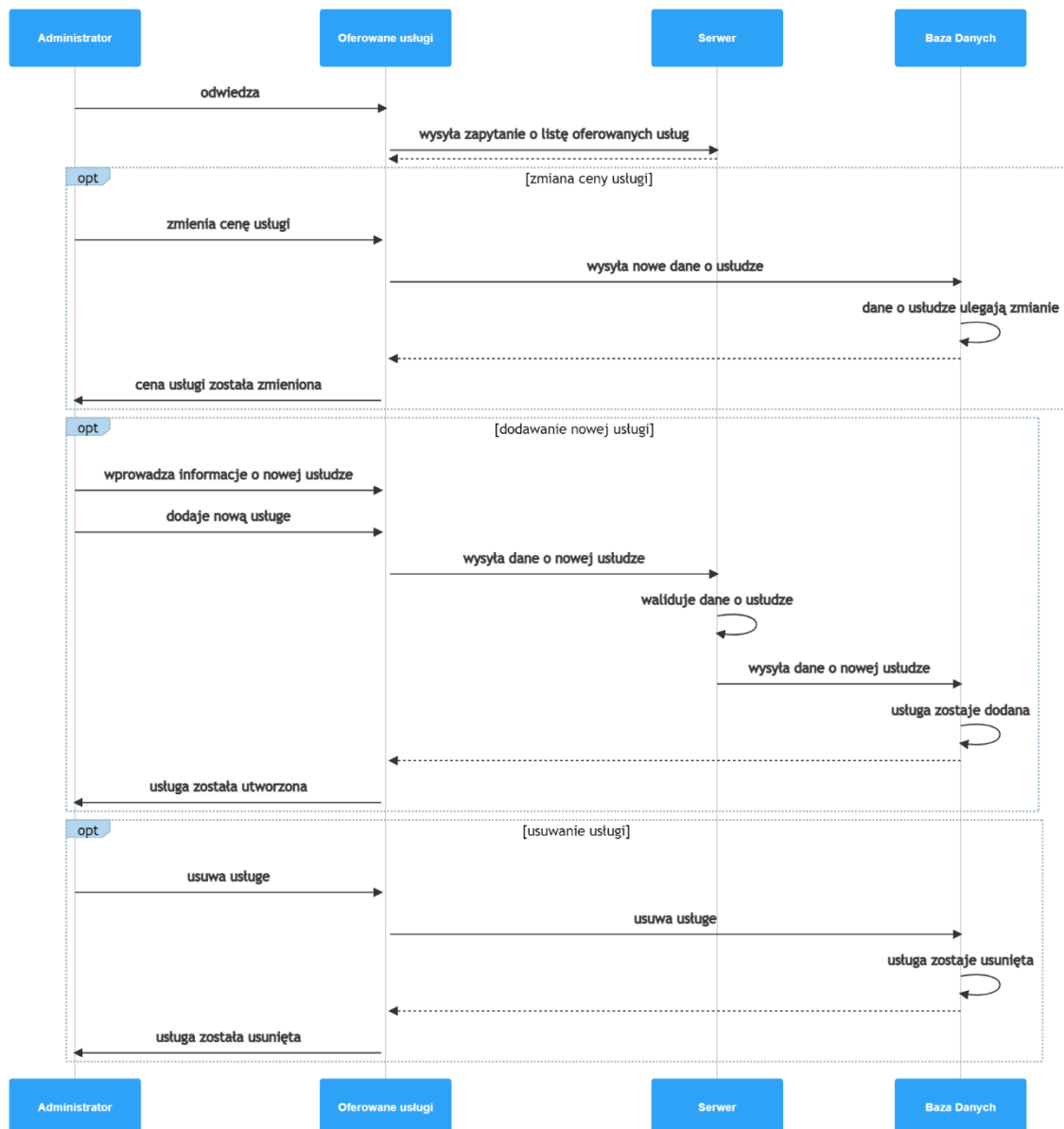


Rysunek 11 Zarządzanie pracownikami

Administrator ma możliwość dodawania nowych pracowników. Po wprowadzeniu danych osobowych i kliknięciu w przycisk 'Dodaj', dane o pracowniku wysyłane są na Serwer i walidowane. Następnie w Bazie danych tworzony jest nowy rekord opisujący pracownika. Z Bazy danych wysyłany jest komunikat zwrotny o poprawnym dodaniu pracownika.

Administrator może również usuwać pracowników. Po kliknięciu w przycisk 'Usuń', pracownik zostaje usunięty z Bazy Danych a na Liście pracowników zostaje wyświetlony komunikat o jego poprawnym usunięciu.

2. Zarządzania ofertą usług i ich cennikiem



Rysunek 12 Zarządzanie usługami

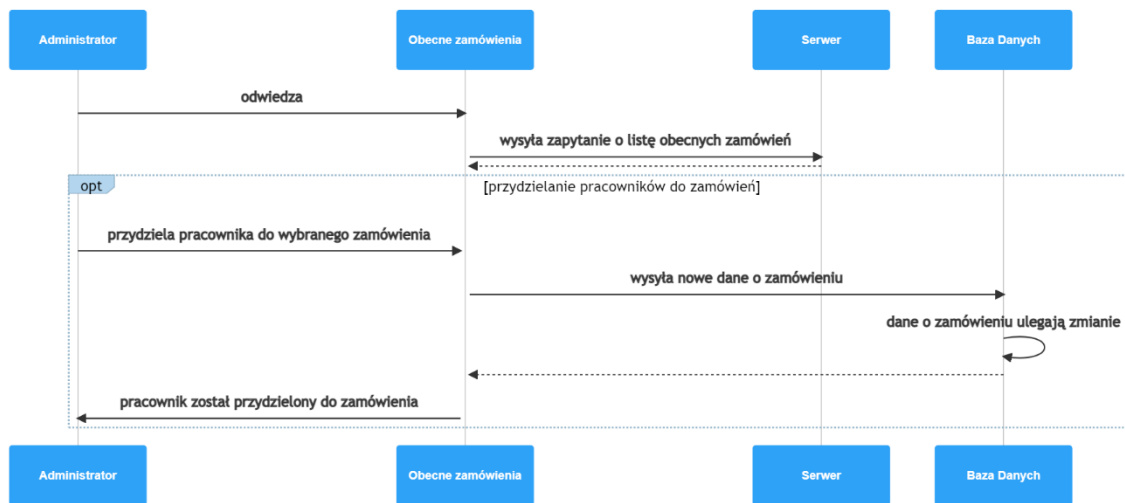
Administrator może również zarządzać usługami.

Zmiana ceny konkretnej usługi powoduje wysłanie nowych danych o usłudze na Serwer a następnie modyfikację jej rekordu w Bazie danych. Następnie, Administrator otrzymuje komunikat o poprawnej zmianie ceny usługi.

Dodawanie nowej usługi działa podobnie. Po walidacji danych o usłudze po stronie Serwera, w Bazie danych tworzony jest rekord nowej usługi a do aplikacji wysyłany jest komunikat o poprawnym utworzeniu usługi.

Logika usuwania usług działa tak samo jak logika usuwania pracowników.

3. Przeglądania wszystkich zamówień i przydzielania Pracowników do poszczególnych zamówień.



Rysunek 13 Zarządzanie zamówieniami przez Administratora

Ponadto, Administrator może zarządzać wszystkimi zamówieniami i przydzielać do nich pracowników. Po przydzieleniu pracownika do konkretnego zamówienia, nowe dane o zamówieniu wysyłane są na Serwer i do Bazy danych. Po poprawnej zmianie przypisanego pracownika, w aplikacji wyświetlony zostaje komunikat.

4. Projekt interfejsu użytkownika

Interfejs użytkownika będzie składać się z kilku podstron. To pozwoli na ułatwione korzystanie z portalu ze względu na podzielenie różnych odpowiedzialności poszczególnych zadań.

Nagłówek strony internetowej zawierać będzie listę odnośników, za pomocą których użytkownik będzie mógł nawigować po aplikacji. W zależności od tego, czy użytkownik jest zalogowany do aplikacji, to lista dostępnych odnośników będzie się różniła. W przypadku, gdy użytkownik nie jest zalogowany do systemu będzie widnieć odnośnik do zalogowania się. W przypadku, gdy użytkownik jest zalogowany – nagłówek będzie wyświetlać odnośnik do strony z panelem do zarządzania zamówieniami, a także odnośnik do wylogowania się użytkownika.



Rysunek 14 Wygląd nagłówka w przypadku, gdy użytkownik nie jest zalogowany



Rysunek 15 Wygląd nagłówka w przypadku, gdy użytkownik jest zalogowany

Na głównej stronie będzie widoczny podstawowy formularz, w którym klient może dokonać wstępnej analizy kosztów wykonania usługi. Formularz ten będzie zawierał pola, w których możemy ustalić liczbę pomieszczeń w budynku, miejsce, w którym dana usługa będzie dokonywana oraz rodzaj usługi. Będą widoczne również podstawowe informacje o ofercie firmy sprzątającej. Każda oferta widoczna na stronie głównej zawierać będzie nazwę usługi, krótki opis oraz szacunkową cenę za wykonanie usługi. Po kliknięciu w jedną z nich system przekieruje nas na stronę, w której możemy dokonać zamówienia danej usługi.

NAGŁÓWEK STRONY

Dokonaj podstawowej analizy kosztów

Liczba pomieszczeń

—

00

+

Miasto


▼

Rodzaj usługi

▼

Analizuj


Nasze usługi



Usługa 1

100zł


Krótki opis usługi



Usługa 2

100zł

Krótki opis usługi



Usługa 3

100zł

Krótki opis usługi


STOPKA

Rysunek 16 Ogólny schemat strony głównej aplikacji

System udostępni możliwość zalogowania się i rejestracji do systemu. W obu przypadkach będą utworzone pola w formularzu, do których należy wprowadzić wymagane dane. Wartości w każdej komórce będą poddawane walidacji. To pozwoli nam uniknąć problemów związanych z nieprawidłowymi danymi.

Strona logowania będzie zawierać panel, w którym będą widniały dwa pola: pole tekstowe, w którym należy wprowadzić nazwę użytkownika, oraz pole tekstowe z maskowaniem znaków, które będzie wykorzystywane do wprowadzenia hasła. Na samym dole tejże strony będzie umieszczony przycisk, który posłuży do zalogowania się użytkownika. Jeżeli logowanie zakończy się powodzeniem, wówczas

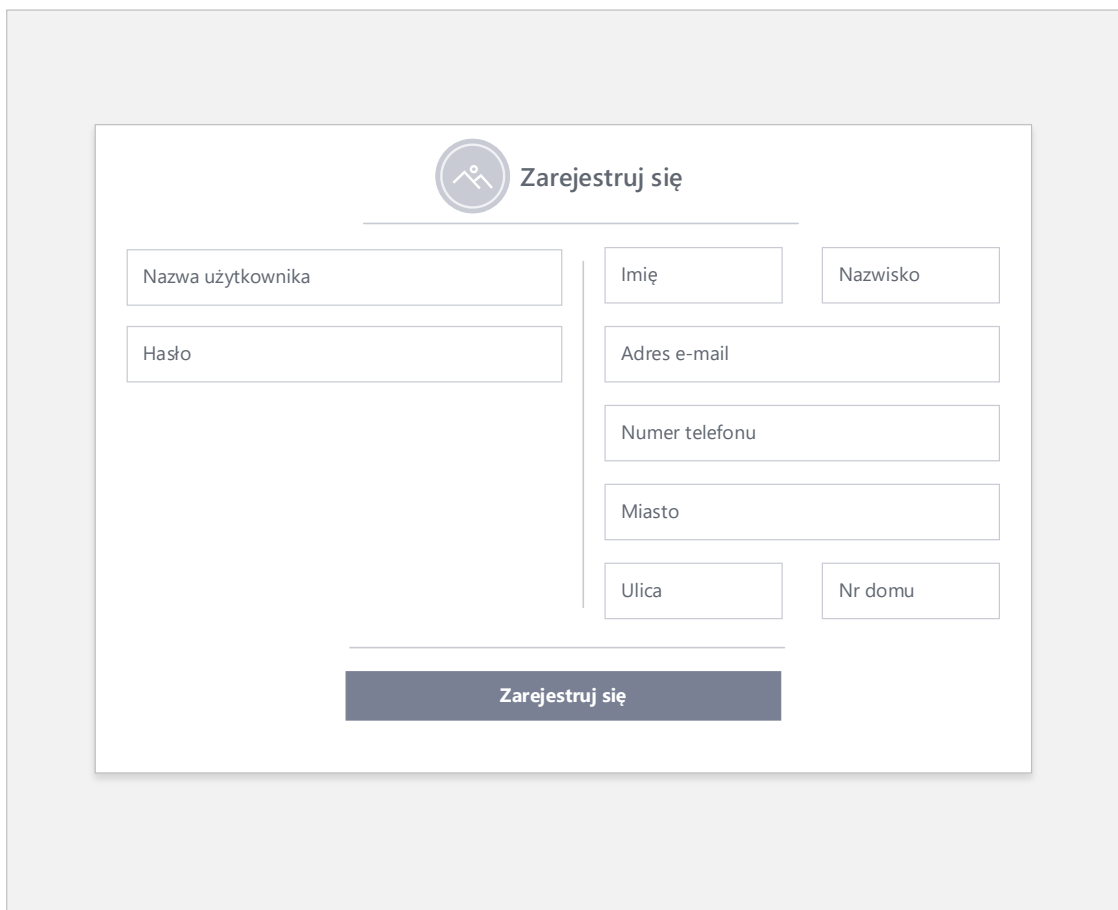
system przekieruje nas na stronę główną aplikacji. Pod danym przyciskiem umieszczony będzie tekst, który zasugeruje użytkownikowi zarejestrowanie się do aplikacji w przypadku, gdy konta jeszcze nie posiada.



The image shows a login form schematic centered on a light gray background. The form is a white rectangle with a thin gray border. At the top left of the form is a circular icon containing a stylized person. To its right is the text 'Zaloguj się'. Below this header, there are two input fields: the first is labeled 'Nazwa użytkownika' and the second is labeled 'Hasło'. Below the input fields is a dark gray button with the text 'Zaloguj się' in white. At the bottom of the form, there is a line of text: 'Jeżeli nie posiadasz konta, [Zarejestruj się](#)'.

Rysunek 17 Schemat przedstawiający wzór formularza logowania

Strona rejestracji natomiast zawierać będzie panel z polami, w których należy wprowadzić następujące dane: nazwę użytkownika, hasło, imię, nazwisko, adres e-mail, numer telefonu oraz adres zamieszkania. Strona rejestracji będzie działała podobnie jak strona logowania, z tą różnicą, że będziemy mieli do dyspozycji zdecydowanie większą liczbę pól do wypełnienia. Po poprawnym zarejestrowaniu system automatycznie zaloguje użytkownika do systemu i przekieruje go na stronę główną portalu.

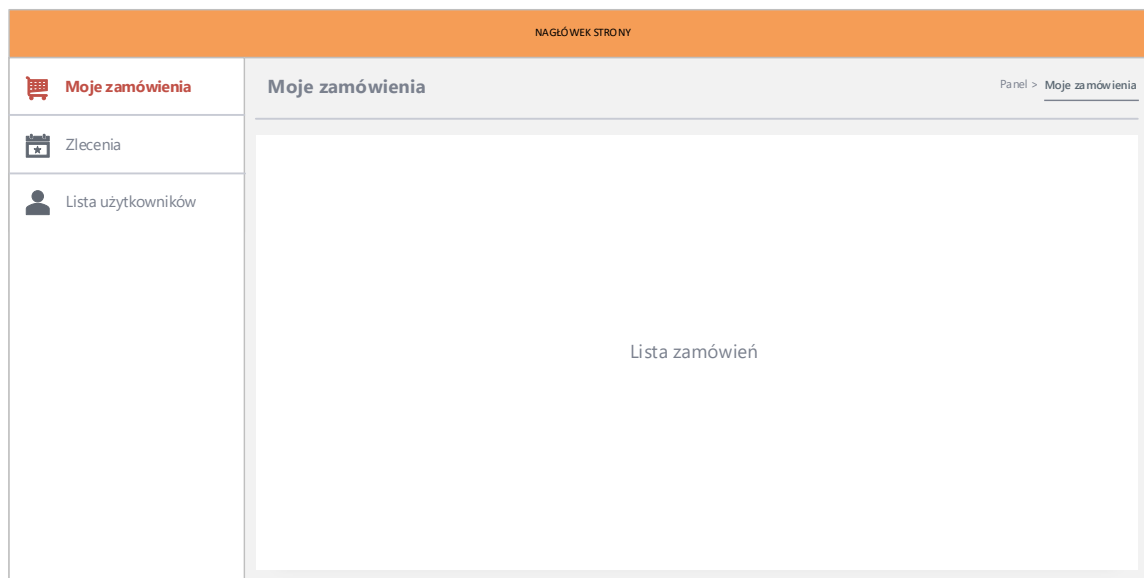


The image shows a registration form titled "Zarejestruj się" (Register) with a circular icon containing a person silhouette. The form is organized into two main columns. The left column contains two input fields: "Nazwa użytkownika" (Username) and "Hasło" (Password). The right column contains several input fields: "Imię" (First Name) and "Nazwisko" (Last Name) at the top, followed by "Adres e-mail", "Numer telefonu" (Phone Number), "Miasto" (City), "Ulica" (Street), and "Nr domu" (House Number). At the bottom of the form is a dark blue button labeled "Zarejestruj się".

Zarejestruj się	
Nazwa użytkownika	Imię Nazwisko
Hasło	Adres e-mail
	Numer telefonu
	Miasto
	Ulica Nr domu
Zarejestruj się	

Rysunek 18 Schemat przedstawiający wzór formularza do rejestracji

Po zalogowaniu się (lub rejestracji) użytkownika użytkownik będzie mógł dokonać zamówienia. Po dokonaniu zamówienia użytkownik będzie mógł przeglądać swoje zamówienia w panelu klienta. Dostęp do panelu będzie możliwy po zalogowaniu się użytkownika. Panel ten będzie dostępny dla każdego zalogowanego użytkownika. W zależności od tego, jaką rolę posiada użytkownik, niektóre opcje nie będą w nim widoczne.



Rysunek 19 Schemat przedstawiający ogólny schemat wyglądu panelu użytkownika

5. Projekt testów

Moduły w poszczególnych warstwach systemu zostaną pokryte testami jednostkowymi oraz zostaną dla nich przeprowadzone testy integracyjne. Następnie system zostanie przetestowany funkcjonalnie, aby zweryfikować zgodność z wymaganiami funkcjonalnymi. Kolejnym krokiem będą testy wydajnościowe oraz testy zgodności sprawdzające zgodność z wymaganiami pozafunkcjonalnymi.

5.1. Testy funkcjonalne

Lp.	Obszar	Testowany scenariusz	Oczekiwany rezultat	Gość	Klient	Pracownik	Administrator
1	Rejestracja konta	Nie wprowadzono wymaganych danych do rejestracji	Komunikat błędu zostaje wyświetlony użytkownikowi. Konto nie zostaje utworzone.		N/A	N/A	N/A
2		Wprowadzono niepoprawne dane w wymagane pola	Komunikat błędu zostaje wyświetlony użytkownikowi. Konto nie zostaje utworzone.		N/A	N/A	N/A
3		Wprowadzono poprawne dane w wymagane pola	Konto zostaje utworzone.		N/A	N/A	N/A
4	Logowanie do systemu	Nie wprowadzono danych logowania	Komunikat błędu zostaje wyświetlony użytkownikowi. Użytkownik nie zostaje zalogowany.	N/A			
5		Nieprawidłowa nazwa użytkownika	Komunikat błędu zostaje wyświetlony użytkownikowi. Użytkownik nie zostaje zalogowany.	N/A			
6		Nieprawidłowe hasło	Komunikat błędu zostaje wyświetlony użytkownikowi. Użytkownik nie zostaje zalogowany.	N/A			
7		Wprowadzono prawidłowe dane logowania	Użytkownik zostaje zalogowany.	N/A			
8	Usługi	Przeglądanie ofertowanych usług	Użytkownik ma dostęp do ofertowanych usług i może się zapoznać z ofertą.				
9		Edytowanie usługi i jej ceny	Usługa zostaje zmodyfikowana. Wszystkie nowe zamówienia są wyceniane na podstawie nowej ceny.	N/A	N/A	N/A	

Lp.	Obszar	Testowany scenariusz	Oczekiwany rezultat	Gość	Klient	Pracownik	Administrator
10		Dodawanie nowej usługi do oferty	Usługa zostaje dodana. Użytkownik ma dostęp do nowej usługi podczas przeglądania oferty i składania zamówienia.	N/A	N/A	N/A	
11		Usuwanie usługi z oferty	Usługa zostaje usunięta. Użytkownik nie ma dostępu do tej usługi podczas przeglądania oferty i składania zamówienia.	N/A	N/A	N/A	
12		Wycenianie usług na podstawie wprowadzonych przez użytkownika danych	Usługa zostaje wyceniona na bazie uzupełnionych przez użytkownika danych. Rezultat zostaje wyświetlony użytkownikowi. Użytkownik może złożyć zamówienie na wycenione wcześniej usługi.				
13	Zamówienia	Tworzenie zamówienia na sprzętanie jednorazowe	Zamówienie zostaje utworzone w systemie.	N/A			
14		Tworzenie zamówienia na sprzętanie cykliczne	Zamówienie zostaje utworzone w systemie.	N/A			
15		Przeglądanie zamówień	Klient ma dostęp do złożonych przez siebie zamówień i ma wgląd w ich szczegóły. Pracownik ma dostęp złożonych przez siebie zamówień oraz do przydzielonych do siebie zamówień. Administrator ma dostęp do wszystkich zamówień złożonych.	N/A			
16		Anulowanie zamówienia	Zamówienie zostaje anulowane i nie jest	N/A			

Lp.	Obszar	Testowany scenariusz	Oczekiwany rezultat	Gość	Klient	Pracownik	Administrator
			dłużej widoczne w systemie.				
17		Przydzielanie zamówienia	Zamówienie zostaje przypisane do pracownika. Status zamówienia zmienia się na <i>Potwierdzone</i> .	N/A	N/A	N/A	
18		Oznaczanie zamówienia jako wykonane	Zamówienie zmienia status w systemie na <i>Wykonane</i> .	N/A	N/A		
19	Zarządzanie użytkownikami	Przeglądanie użytkowników	Administrator ma dostęp do danych wszystkich użytkowników w systemie (w tym klientów) i może wyświetlić ich szczegóły. Administrator może również edytować ich role w systemie.	N/A	N/A	N/A	
20		Dodawanie pracownika	Rola pracownika zostaje dodana do konkretnego użytkownika. Użytkownik uzyskuje dostęp do funkcjonalności systemu dostępnych dla pracowników.	N/A	N/A	N/A	
21		Usuwanie pracownika	Rola pracownika zostaje usunięta z konkretnego użytkownika. Użytkownik traci dostęp do funkcjonalności systemu dostępnych dla pracowników.	N/A	N/A	N/A	

5.2. Testy zgodności

Lp.	Obszar	Testowany scenariusz	Oczekiwany rezultat
1	RODO i pliki Cookies	Pierwsze wejście na stronę	Informacja o przechowywaniu plików Cookies na urządzeniu użytkownika jest widoczna na górze strony. Przycisk potwierdzający zrozumienie informacji jest widoczny i dostępny dla użytkownika.
2		Wejście na stronę bez wcześniejszego potwierdzenia zrozumienia informacji dot. plików Cookies	Informacja o przechowywaniu plików Cookies na urządzeniu użytkownika jest widoczna na górze strony. Przycisk potwierdzający zrozumienie informacji jest widoczny i dostępny dla użytkownika.
3		Wejście na stronę po potwierdzeniu zrozumienia informacji dot. plików Cookies	Informacja o przechowywaniu plików Cookies na urządzeniu użytkownika nie jest widoczna.

5.3. Testy wydajnościowe

Lp.	Obszar	Testowany scenariusz	Oczekiwany rezultat
1	Wydajność	Prędkość ładowania strony z poziomu narzędzi developerskich w przeglądarce Internetowej	Czas odpowiedzi serwera nie przekracza 500 ms.
2		Czas odpowiedzi serwera na żądanie http	Czas odpowiedzi serwera nie przekracza 500 ms.