

Wirtualny sprzedawca

(Nowa część dokumentacji znajduje się od 10 strony)

Spis treści

Dziedzina problemowa	2
Cel	2
Zakres odpowiedzialności systemu	2
Użytkownicy systemu	2
Wymagania użytkownika	2
System przechowuje informacje dotyczące	2
Oczekuje się, że system będzie umożliwiał użytkownikom	4
System powinien spełniać następujące warunki	4
Wymagania funkcjonalne	5
Diagram przypadków użycia	5
Diagram aktywności	6
Diagram kontekstowy	7
Analityczny diagram klas	8
Wymagania нефункционалне	9
Przypadki użycia	10
Powielony diagram aktywności przypadków użycia	10
Scenariusz przypadku „Utworzenie zamówienia” wraz z przypadkiem „Dodanie karty płatniczej”	11
Diagram aktywności	12
Diagram stanu	12
Diagram interakcji	13
Projekt interfejsu dla przypadku użycia	14
Skutki analizy dynamicznej	16
Poprawiony diagram analityczny	16
Omówienie skutków	17
Decyzje projektowe	17
Diagram projektowy	18
Opis przyszłej ewolucji systemu	19
Słownik	19

Dziedzina problemowa

Z powodu kwarantanny wiele lokali zostało zamkniętych, a pracownicy nie mogą wypełniać swoich obowiązków. Z tego powodu zaistniała potrzeba na umożliwienie lokalom dalszego funkcjonowania spełniając normy bezpieczeństwa, takie jak dostawy bezkontaktowe.

Cel

Przyspieszenie transakcji oraz ograniczenie kontaktu z innymi ludźmi poprzez zastąpienie kasjerów stacjonarnych aplikacją w barach na wynos, stacjach benzynowych i wszędzie tam gdzie opłacamy usługę lub kupujemy produkt nie musząc wchodzić w kontakt ze sprzedawcą. Pozwala to utrzymać działanie restauracji, barów oraz drobnych sklepów spożywczych w okresie ścisłej kwarantanny.

Zakres odpowiedzialności systemu

System powinien pozwalać użytkownikom dokonywać zakupów, tworzyć zamówienia (uwzględniając produkty lub usługi) oraz opłacać je, a pracownikom potwierdzać zamówienia (które mogą zostać zrealizowane).

Użytkownicy systemu

- Pracownicy lokalu:
 - Operator
 - Realizator
- Edytor
- Użytkownik zarejestrowany
- Gość

Wymagania użytkownika

System przechowuje informacje dotyczące

- Klientów
- Pracowników lokalu
- Audytorów
- Lokalów
- Ofert
- Zamówień

Klient	Dane personalne	imię
		nazwisko
		numer telefonu
		adres email
	Dane logowania	login
		hasło
	Dane płatnicze	numer karty
		typ karty
		CVC
		data wygaśnięcia
	Historia zamówień	zamówienia
Pracownik lokalu	Dane personalne	imię
		nazwisko
		numer telefonu
		adres email
	Dane logowania	login
		hasło
Edytor	Dane personalne	imię
		nazwisko
		numer telefonu
		adres email
	Dane logowania	login
		hasło
Lokal	Dane spółki	nazwa
		branża
	Dane lokalu	ulica
		numer
		kod pocztowy
		miejsowość
		oferta
Oferta	Dane oferty	edytor
		audytor
		data utworzenia
		data opublikowania
	Dane pozycji	nazwa
		cena
Zamówienie	Dane realizacyjne	data utworzenia
		data realizacji
		status
		operator
		realizator
		klient
		lokal
	Dane oferty	pozycje oferty
		kwota

Oczekuje się, że system będzie umożliwiał użytkownikom

- Przeglądanie oferty – wszyscy
- Rejestrację – gość
- Dodanie karty – klient
- Utworzenie zamówienia – klient
- Zgłaszanie niezgodności zamówienia – klient
- Odczytywanie historycznych zamówień – klient
- Potwierdzenie realizacji zamówienia – pracownik lokalu
- Zarządzanie zamówieniami – operator
- Generowanie raportów sprzedaży – operator
- Zarządzanie pracownikami lokalu – operator
- Zarządzanie ofertą – edytor

System powinien spełniać następujące warunki

- Dla serwera:
 - Obsługuje jednocześnie minimum 100 zapytań
 - Tworzy regularnie kopie zapasowe
 - Dane wrażliwe są szyfrowane
 - Podczas awarii powinien być naprawiony w ciągu 24h
- Dla klienta mobilnego:
 - Logo firmy w lewym górnym rogu
 - Możliwość wyboru szaty graficznej
 - Regularne aktualizacje bezpieczeństwa
 - Aktualizacje wdrażane stopniowo
 - GUI w wielu językach

Wymagania funkcjonalne

Diagram przypadków użycia

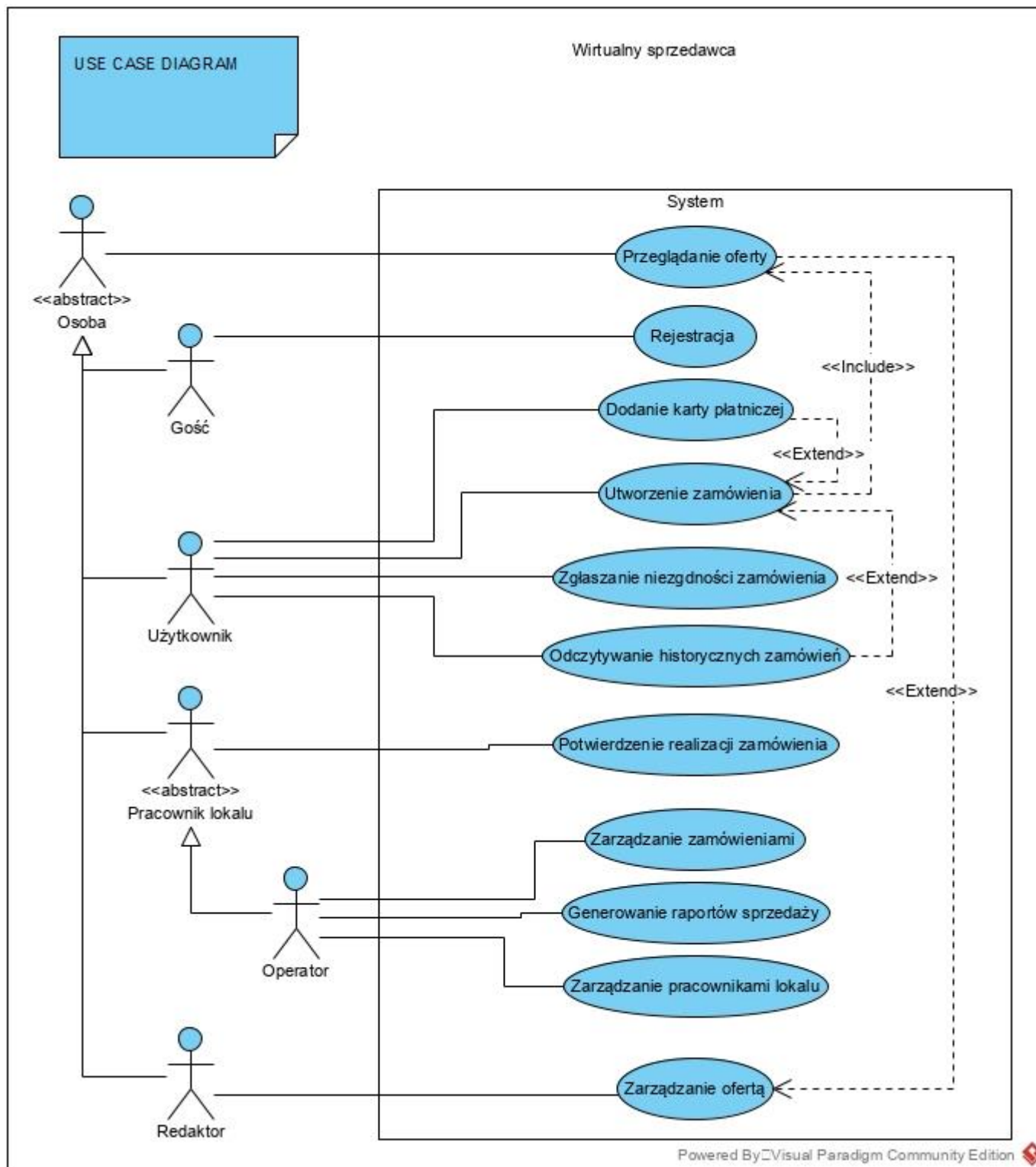
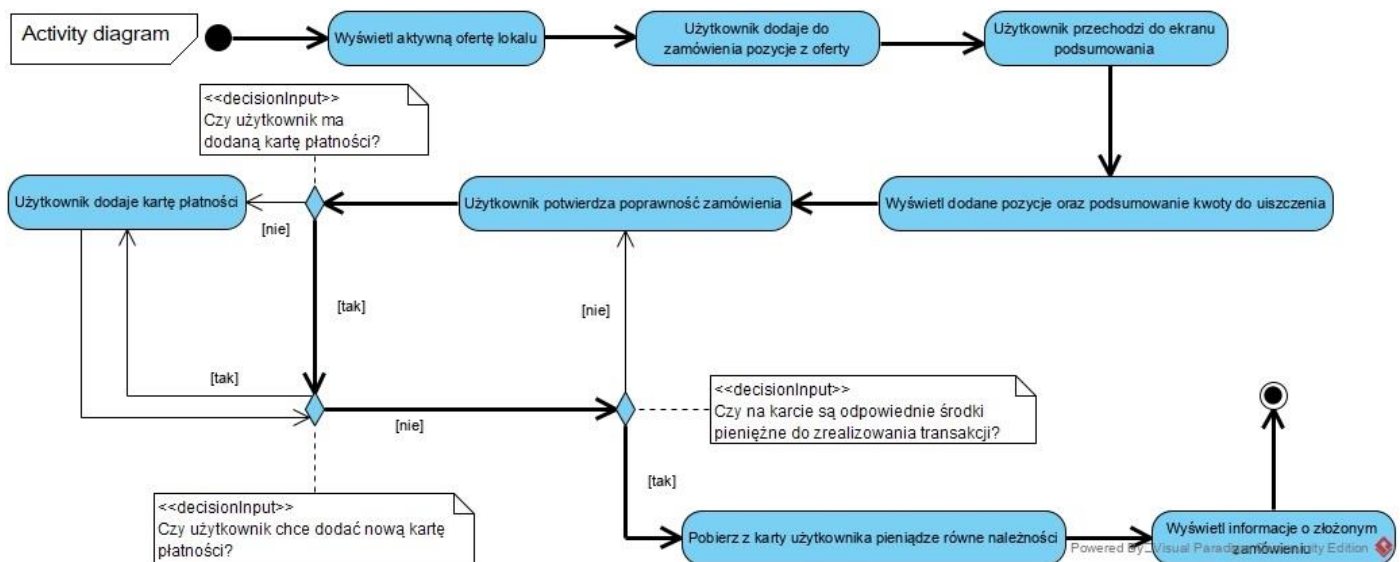


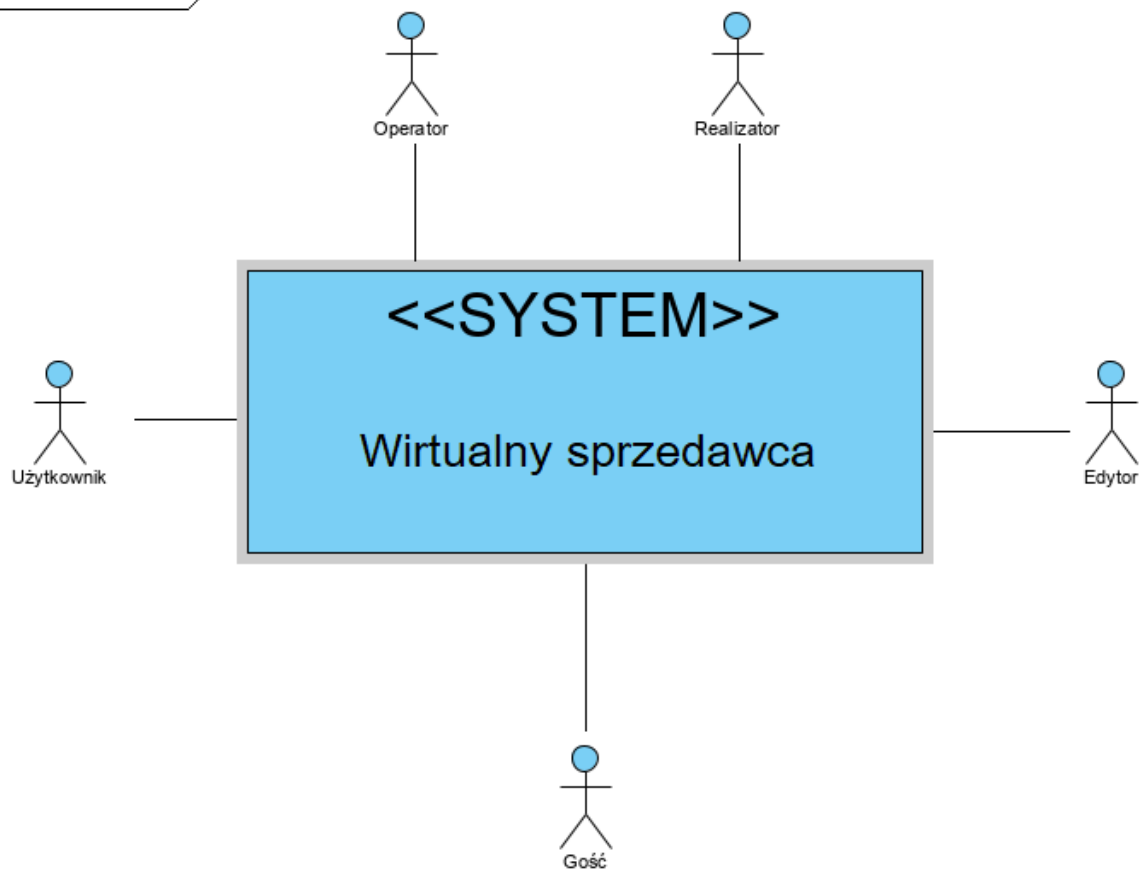
Diagram aktywności



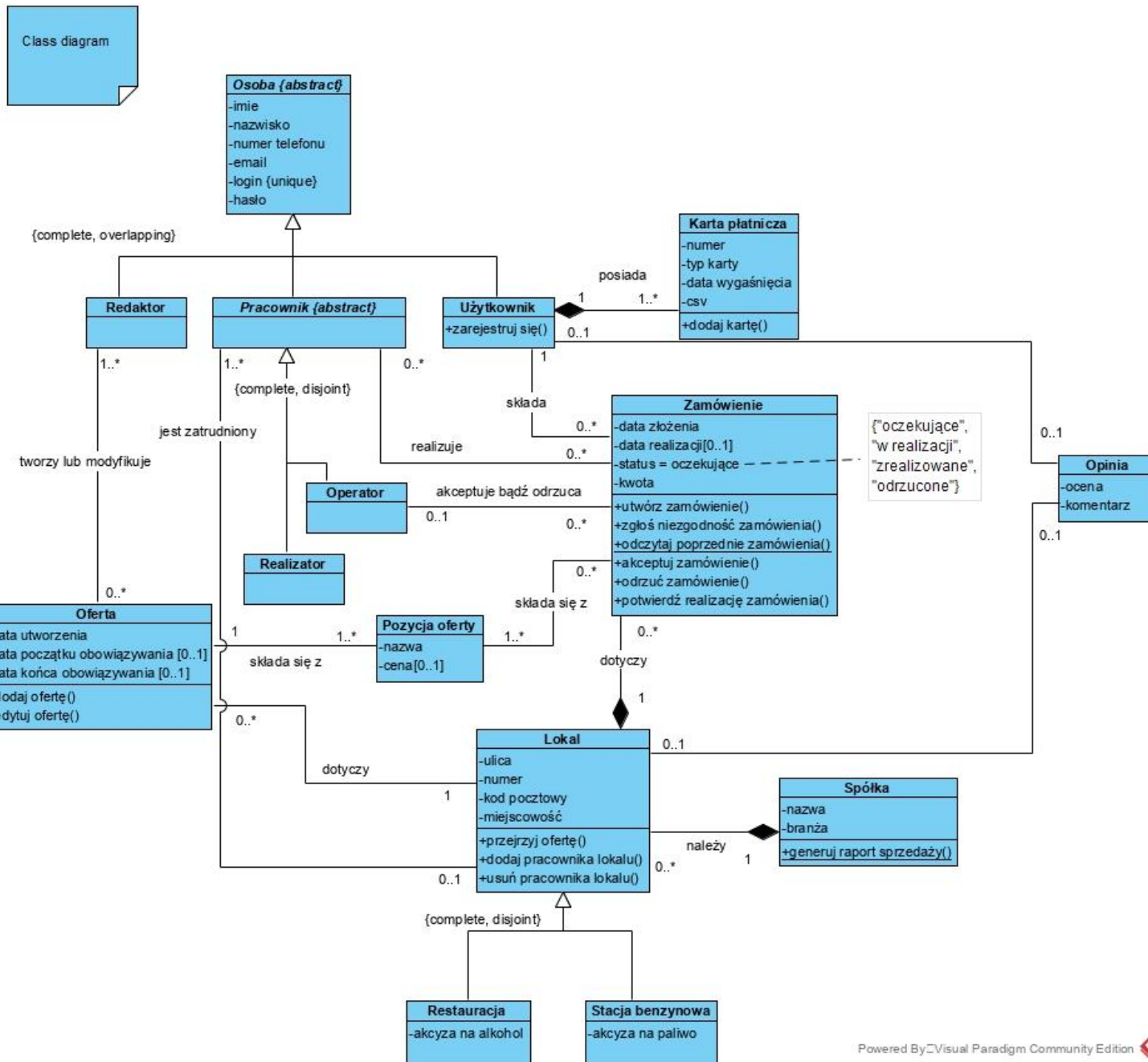
Nazwa przypadku użycia	Utworzenie zamówienia
Warunek początkowy	Użytkownik jest zalogowany oraz posiada co najmniej jedną kartę płatniczą Użytkownik znajduje się w zasięgu lokalu
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Przypadek użycia zostaje uruchomiony przez użytkownika 2. System wyświetla aktywną ofertę danego lokalu 3. Użytkownik dodaje do swojego zamówienia pozycje z oferty 4. Użytkownik kończy dodawanie pozycji poprzez przejście do ekranu podsumowania 5. System wyświetla aktualny koszyk użytkownika w formie podsumowania, wraz z sumą kwoty do uiszczenia 6. Użytkownik potwierdza wprowadzone pozycje 7. System pobiera z karty użytkownika pieniądze za zamówienie 8. System informuje o złożeniu zamówienia
Alternatywny przepływ zdarzeń	<ol style="list-style-type: none"> 6a. Użytkownik dodaje nową kartę płatności 6b. Użytkownik wybiera kartę z jakiej chce dokonać opłaty za zamówienie 7a. Na karcie nie ma dostępnych środków, system wyświetla odpowiednią informację i cofa przepływ zdarzeń do punktu 5
Zakończenie	W dowolnej chwili
Warunek końcowy	W przypadku utworzenia zamówienia, do systemu wprowadzone zostaje zgłoszenie wraz z zawieranymi pozycjami

Diagram kontekstowy

Diagram kontekstowy



Analityczny diagram klas

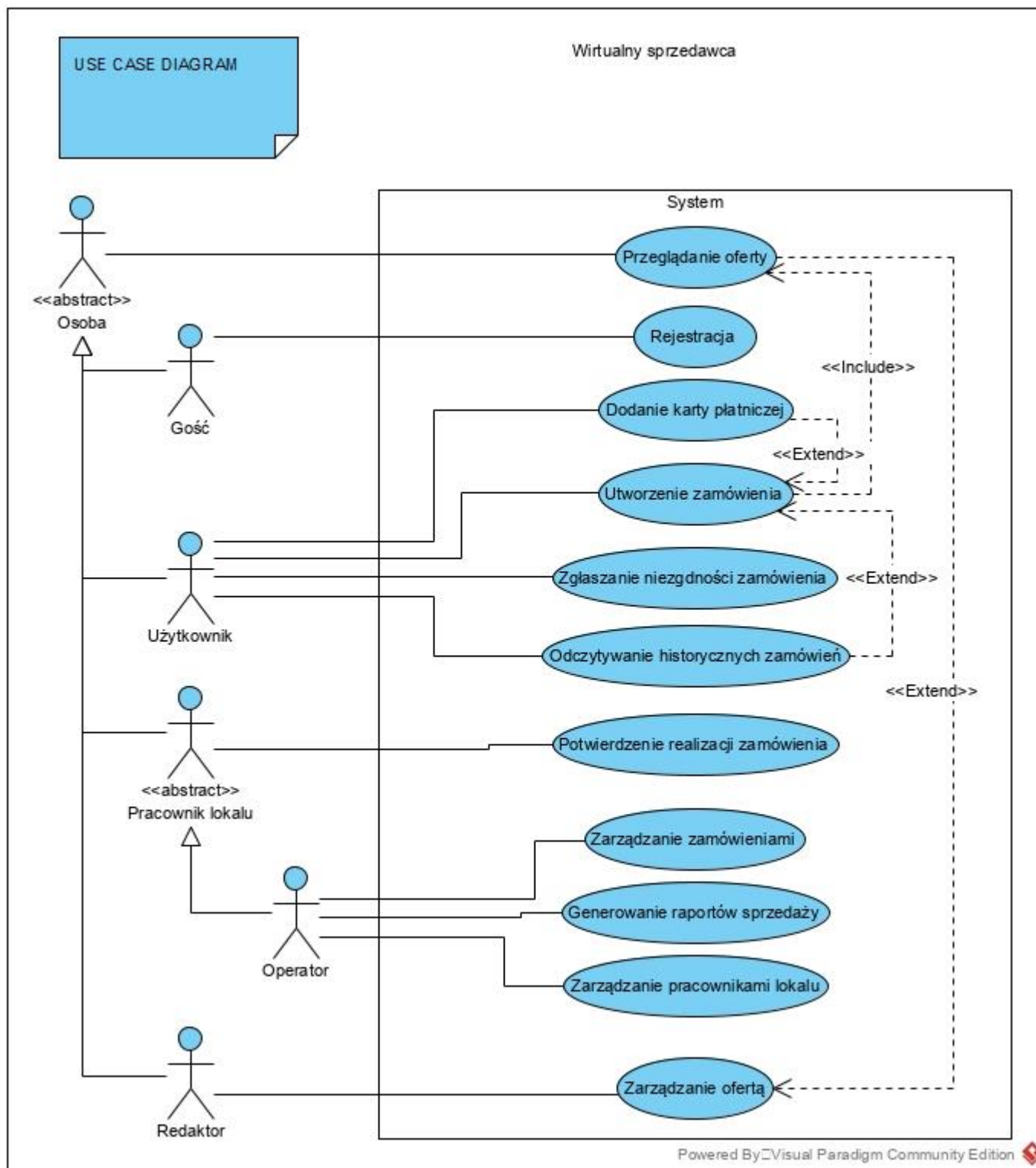


Wymagania niefunkcjonalne

- Dla serwera:
 - Obsługuje jednocześnie minimum 100 zapytań:
 - Baza danych oparta na MSSQL a całość na dwóch oddzielnych maszynach z Windows Server 2016 lub nowszym. Maszyny o minimalnej konfiguracji 8 GB RAM i 16 rdzeni o taktowaniu 2 GHz
 - Tworzy regularnie kopie zapasowe:
 - Co tydzień pełna kopia
 - Codziennie kopia różnicowa
 - Dane wrażliwe są szyfrowane:
 - Algorytm SHA256
 - Podczas awarii powinien być naprawiony w ciągu 24h
 - Spełniana jest norma niezawodności PN-IEC 60300
- Dla klienta mobilnego:
 - Logo firmy w lewym górnym rogu:
 - W formie przycisku menu
 - Możliwość wyboru szaty graficznej:
 - Dostępne dwa motywy, jasny oraz ciemny
 - Regularne aktualizacje bezpieczeństwa:
 - Aplikacja spełnia normy bezpieczeństwa ISO/IEC 27001
 - Aktualizacje wdrażane stopniowo:
 - Aktualizacja dziennie pokrywa maksymalnie 10% wszystkich użytkowników
 - GUI w wielu językach:
 - Polski
 - Angielski
 - Rosyjski

Przypadki użycia

Powielony diagram aktywności przypadków użycia



Scenariusz przypadku „Utworzenie zamówienia” wraz z przypadkiem „Dodanie karty płatniczej”

Nazwa przypadku użycia	Utworzenie zamówienia
Warunek początkowy	Użytkownik jest zalogowany oraz posiada co najmniej jedną kartę płatniczą Użytkownik znajduje się w zasięgu lokalu
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Przypadek użycia zostaje uruchomiony przez użytkownika 2. System wyświetla aktywną ofertę danego lokalu 3. Użytkownik dodaje do swojego zamówienia pozycje z oferty 4. Użytkownik kończy dodawanie pozycji poprzez przejście do ekranu podsumowania 5. System wyświetla aktualny koszyk użytkownika w formie podsumowania, wraz z sumą kwoty do uiszczenia 6. Użytkownik potwierdza wprowadzone pozycje 7. System pobiera z karty użytkownika pieniądze za zamówienie 8. System informuje o złożeniu zamówienia
Alternatywny przepływ zdarzeń	<p>6a. Użytkownik dodaje nową kartę płatności</p> <p>6b. Użytkownik wybiera kartę z jakiej chce dokonać opłaty za zamówienie</p> <p>7a. Na karcie nie ma dostępnych środków, system wyświetla odpowiednią informację i cofa przepływ zdarzeń do punktu 5</p>
Zakończenie	W dowolnej chwili
Warunek końcowy	W przypadku utworzenia zamówienia, do systemu wprowadzone zostaje zgłoszenie wraz z zawieranymi pozycjami

Diagram aktywności

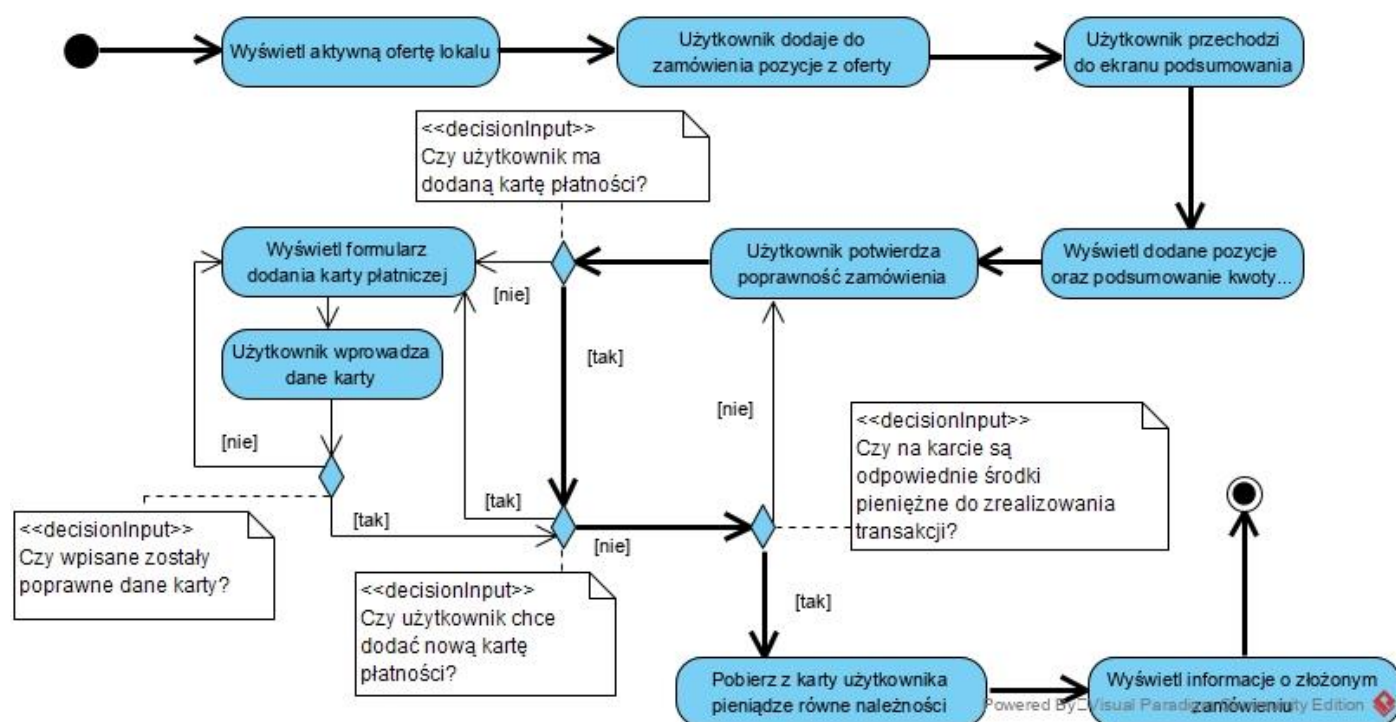


Diagram stanu

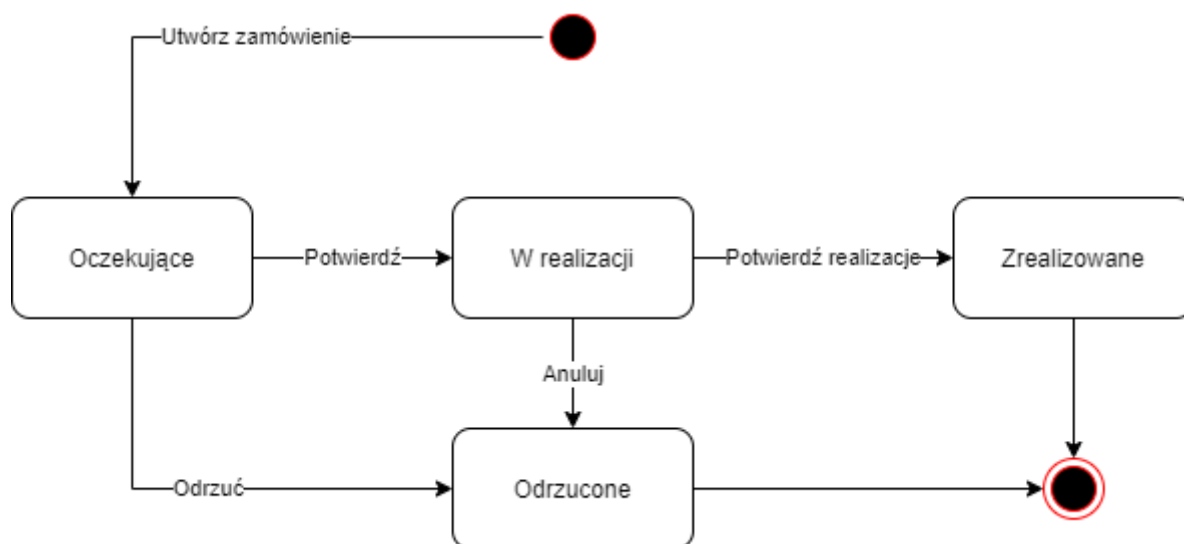
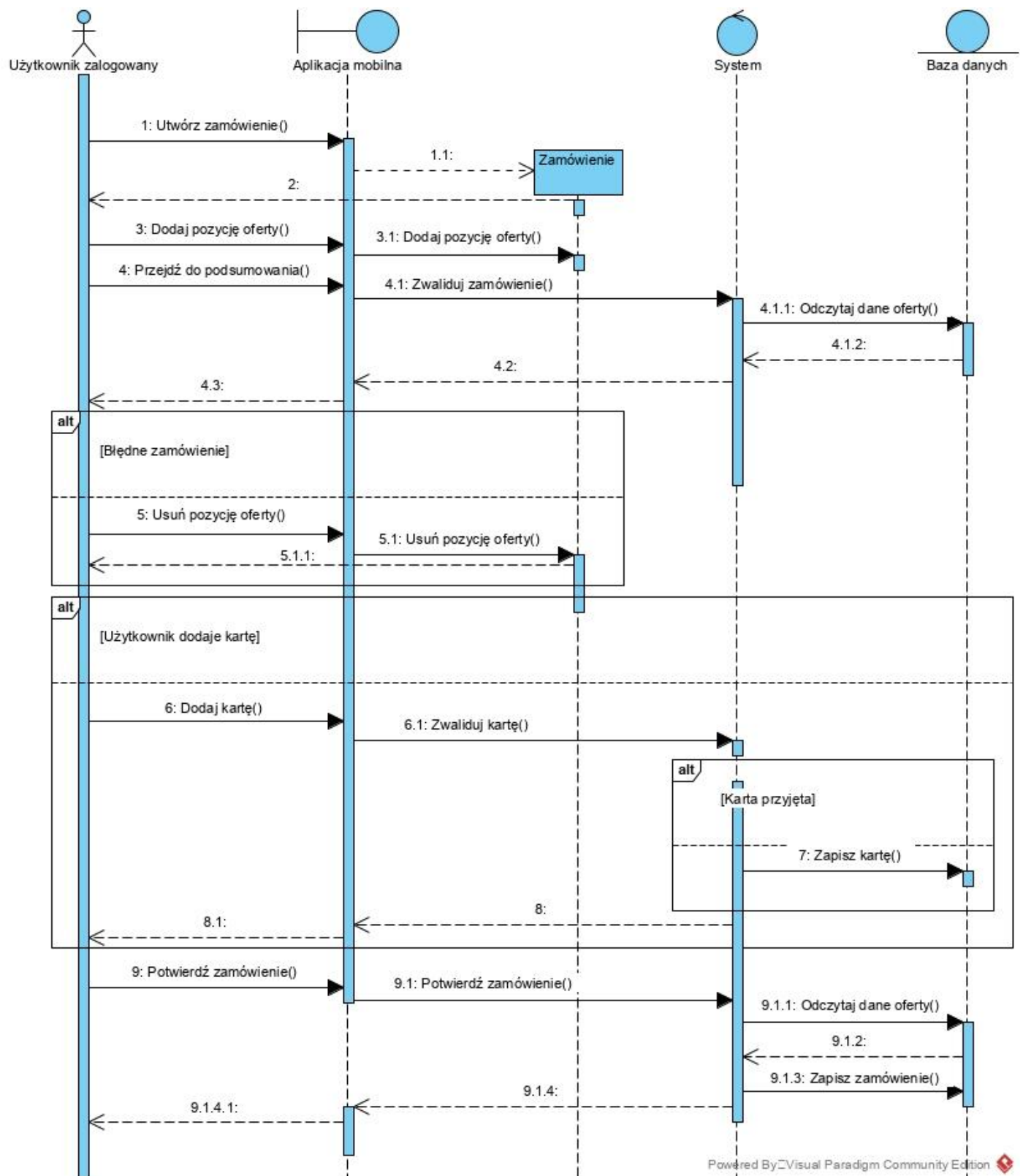
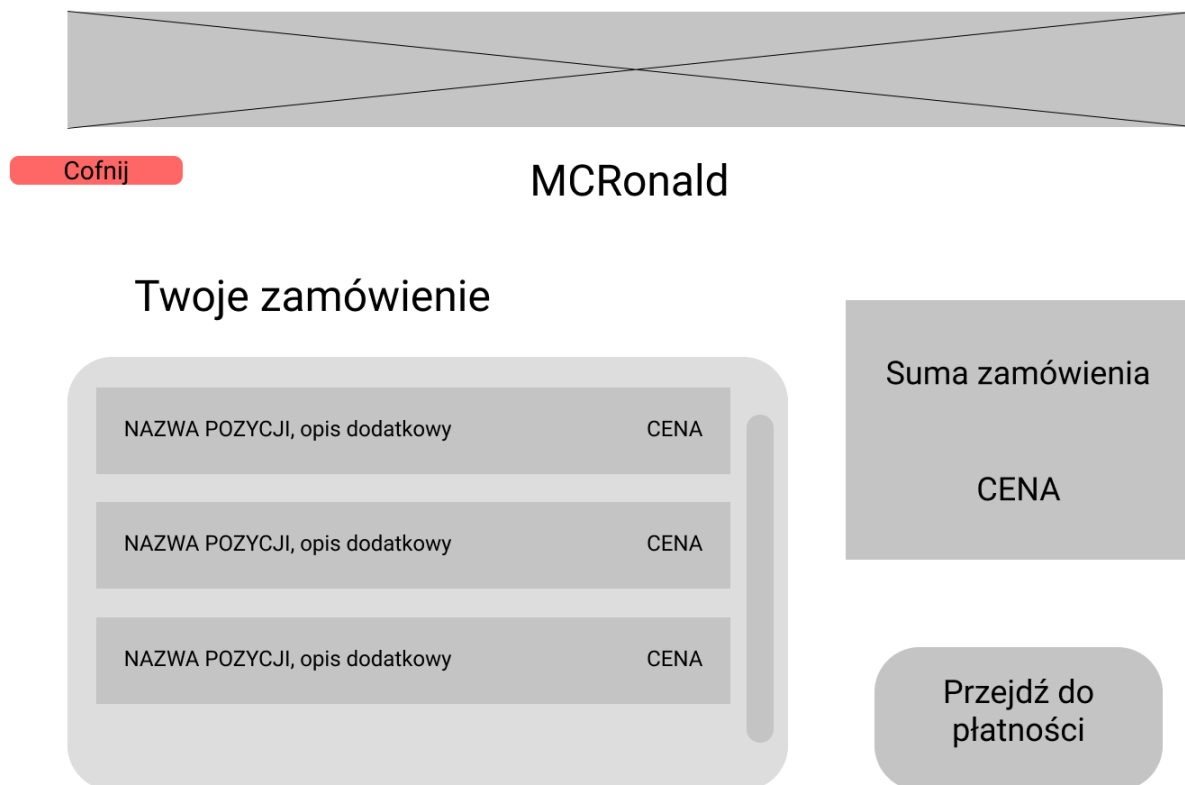
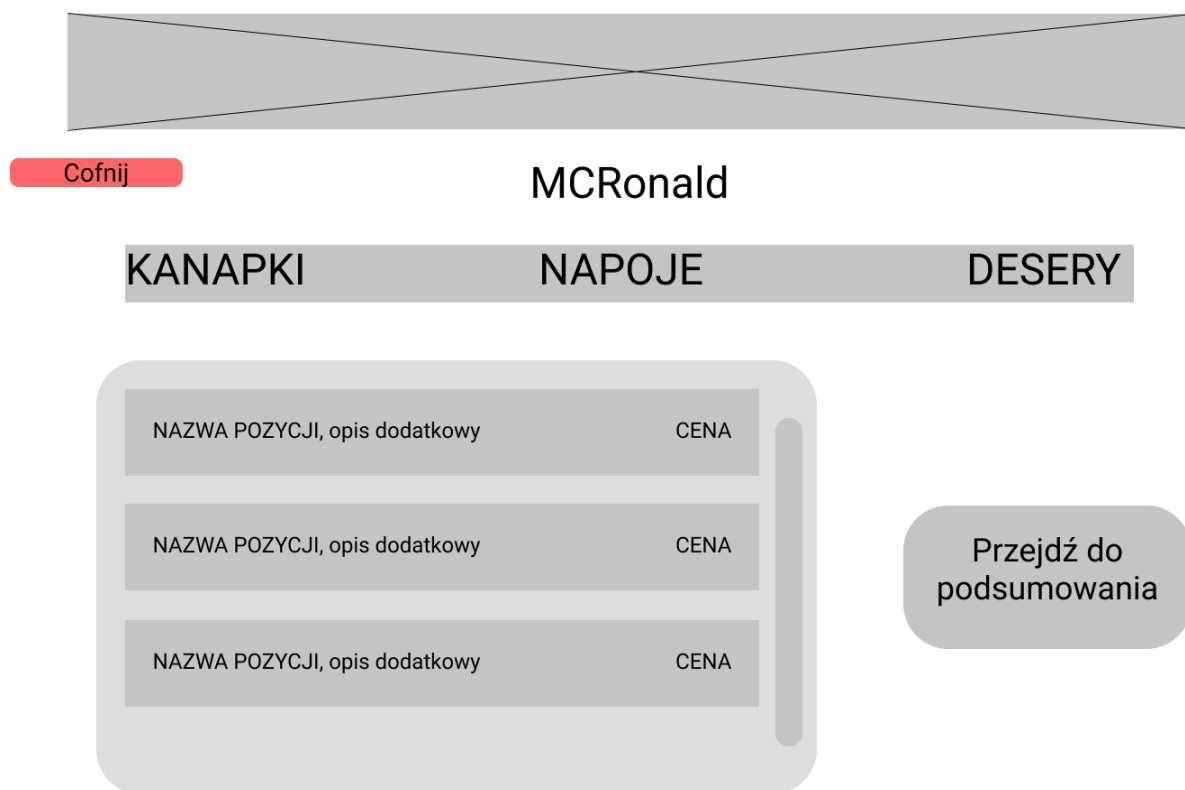
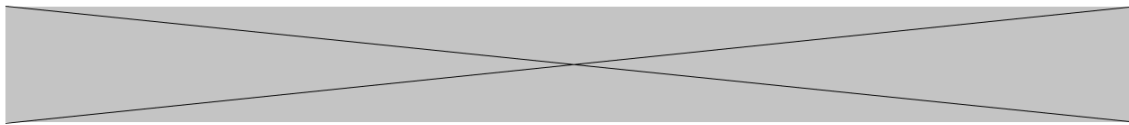


Diagram interakcji







Cofnij

Wybierz kartę z której chcesz
dokonać płatności

KARTA VISA XXXXXX

Dodaj nową kartę +

Dane kontaktowe w
sprawie zamówienia

Imię Nazwisko
mail@mail.mail
123-456-789

XX ZŁ

KUP

Poprawiony diagram analityczny

Class diagram



Omówienie skutków

Po dokładnym przeanalizowaniu scenariusza okazało się, że w projekcie brakuje bardzo ważnej metody w klasie Zamówienie – walidowania. Metoda jest bardzo ważna, ponieważ uruchamiana jest aż dwa razy zanim zamówienie zostanie zapisane w bazie danych.

Zabrakło również metody dodawania pozycji oferty do zamówienia, co znacząco ułatwiło by implementację projektu.

Decyzje projektowe

Do implementacji wykorzystam język Java, jako bazę do przechowywania ekstensji popularną i darmową bazę danych MySQL. Do komunikacji w Javie z bazą danych wykorzystam bibliotekę Hibernate.

W celu zastosowania graficznego interfejsu wykorzystam zestaw narzędzi JavaFX, pozwalający w łatwy sposób projektować skalowane interfejsy, z oprogramowanymi przyciskami.

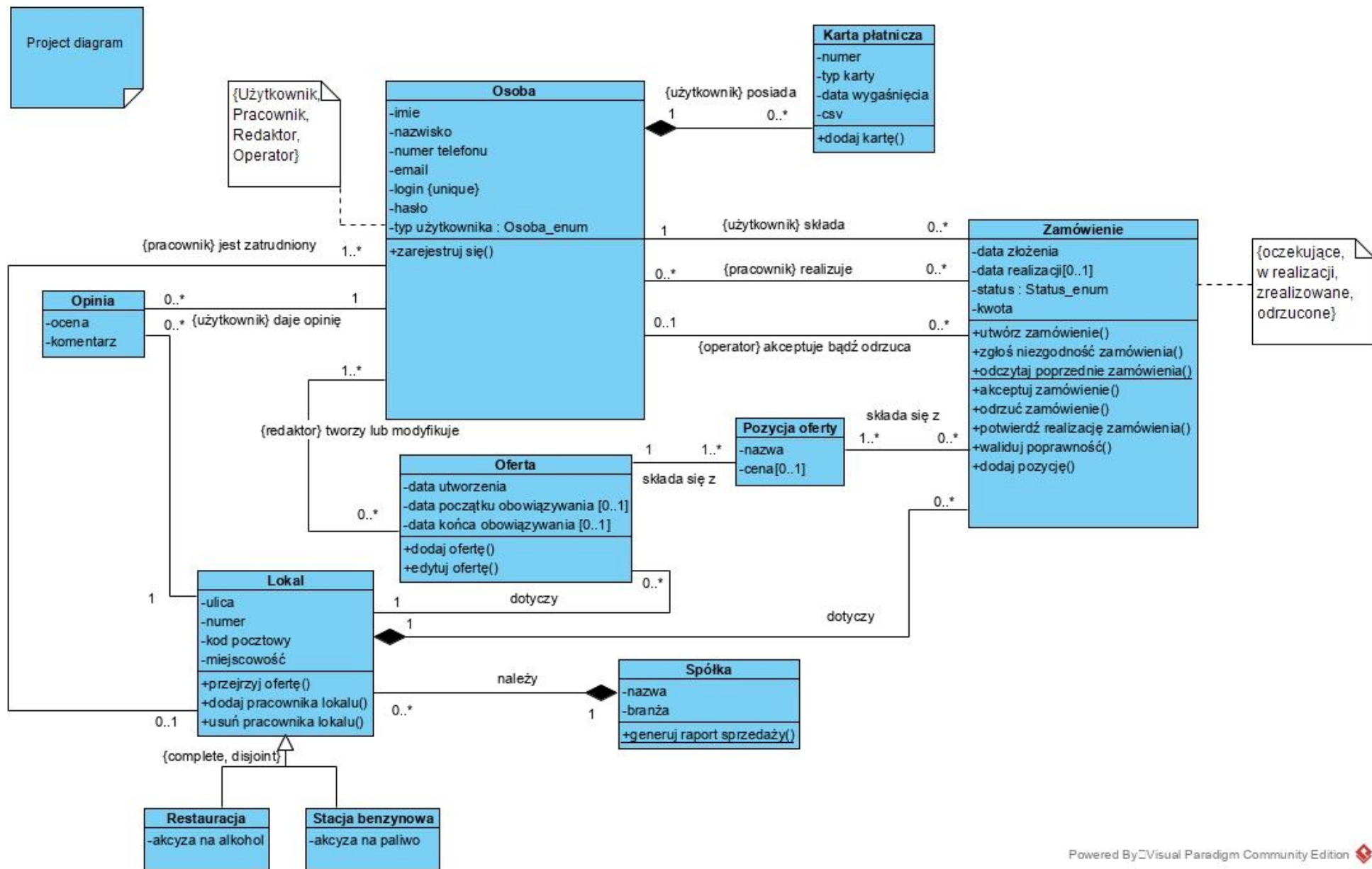
Ze względu na ograniczenia narzucone przez wybrany język programowania oraz sposób przechowywania informacji w bazach relacyjnych, klasy projektu odpowiadające za użytkowników systemu musiały zostać spłaszczone do jednej klasy – Osoba.

Overlapping wymusił skorzystanie ze sposobu implementacji opartym na enumach, czyli każdy obiekt klasy Osoba przypisaną ma listę z wartościami typu enum określającymi typ użytkownika. Typ użytkownika będzie sprawdzany każdorazowo, kiedy użytkownik będzie wykonywał akcję, w celu wyeliminowania sytuacji w której na przykład zwykłemu użytkownikowi wyświetlone zostaną informacje przeznaczone dla pracownika.

Ze względu na dodatkowe dziedziczenie disjoint pomiędzy operatorem a realizatorem, zostanie zastosowane ograniczenie, polegające na zablokowaniu możliwości dodania dwóch ról z tego dziedziczenia do jednej osoby.

Status zamówienia również zostanie zaimplementowany jako typ enum.

Diagram projektowy



Opis przyszłej ewolucji systemu

Jeśli projekt zostanie zaakceptowany przez społeczność, planowany jest rozwój w stronę:

- Większa ilość języków
- Możliwość wybrania między motywami ciemnym i jasnym
- Dodanie typu użytkownika administrator – zajmowałby się administrowaniem spółki, jej lokalami i pracownikami wewnątrz nich

Słownik

Oferta: Spis pozycji składających się na ofertę danego lokalu, ofert dla jednego lokalu może istnieć kilka ale tylko jedna jest aktywna.

Operator: Pracownik o podwyższonych uprawnieniach, kierownik lokalu lub oddelegowany przez niego pracownik. Jego zadaniem jest akceptowanie bądź odrzucanie zamówień wysyłanych przez klientów, oraz wybieranie aktywnej oferty.

Redaktor (Edytor w starej części dokumentacji): Użytkownik którego zadaniem jest utworzenie oraz modyfikacja ofert lokali.