

*Rwrc*

# **Projekt systemu wsparcia sprzedaży**

# Opis systemu:

## 1. Cel projektu

Celem projektu jest opracowanie systemu komputerowego wspomagającą sprzedaż małych jak i dużych sklepów czy hurtowni w tym też dla zespołu sklepów o wspólnym magazynie centralnym. System ma za zadanie przechowywać informacje o towarach i usługach, klientach oraz wystawiać faktury.

## 2. Opis systemu

System ten będzie przechowywał podstawowe informacje niezbędne do prawidłowego funkcjonowania sklepu/hurtowni zapewniając jednocześnie wygodne graficzne narzędzie do zarządzania nim. Baza gromadzić będzie informacje o produktach (ich stanie, cenie itd.), kontrahentach (imieniu bądź nazwie firmy i podobnych), dostawach i dostawcach (ilość dostarczanych produktów, produkty oferowane przez dostawcę), wystawionych fakturach oraz ogólnych finansach firmy. System za pomocą graficznego programu będzie umożliwiał wydruk dokumentów zakupu takich jak paragon czy faktura oraz raportów okresowych.

## 3. Użytkownicy systemu

**Klient** ma bardzo ograniczony dostęp do systemu, większość operacji dokonuje za pośrednictwem sprzedawcy.

Klient przychodząc do sklepu/hurtowni ma możliwość dokonania anonimowego zakupu ( o ile polityka firmy prowadzące sprzedaż na to zezwala) albo poddać się autoryzacji podając swoje niepowtarzalny numer identyfikacyjny-ID (weryfikując je w zależności wymagań firmy prowadzącej sprzedaż np. kartą klienta , czy na słowo) otrzymując możliwość otrzymywania rabatów i korzystania z promocji specjalnych. Klient otrzymuje ID podczas rejestracji, którą może wykonać podczas dowolnego zakupu lub w każdym innym momencie działania punktu sprzedaży. Podczas zapisu musi podać:

- ♣ W przypadku rejestracji jako osoba prywatna  
Imię, nazwisko, pesel, adres, telefon kontaktowy, e-mail
- ♣ W przypadku rejestracji jako działalność gospodarcza (firma)  
Nazwa działalności ,NIP, REGON, adres, telefon kont. ,e-mail

Klient ma możliwość przeglądania oferty firmy za pośrednictwem specjalnie do tego celu przygotowanej strony internetowej i dokonać zamówienie po wcześniejszej autoryzacji (zalogowaniu na tejże stronie).

**Sprzedawca** ma za zadanie kompleksowej obsługi klienta. W tym celu może:

- wyszukiwać towar,
- dokonywać transakcji,
- wydawać towar,
- wprowadzać nowych kontrahentów.
- Przyjmować zwroty
- Wydawać dokumenty zakupu (paragon/faktura)
- Dokonywać korekty wcześniej wystawionych dokumentów zakupu

**Magazynier** jest odpowiedzialny za wprowadzanie nowych towarów, dodawanie dostawców(będącym w systemie zapisanych razem z klientami jako kontrahenci) i dodawanie dostaw.

**Administrator** ma dostęp do wszystkich funkcji dostępnych innym użytkownikom plus dodatkowo dodawać nowy towar do oferty, może też bezpośrednio edytować stan magazynowy(bez wystawiania korekt, dokumentów sprzedaży czy wprowadzania dostaw), „usuwać”(oznaczać jako nieważne i pomijane przez system) błędne transakcje oraz decydować o rabatach czy promocjach. W niedużych punktach sprzedaży, w których jest tylko jeden pracownik wszystko jest zarządzane z konta administratora (administrator pełni funkcje sprzedawcy i magazyniera).

#### **4. Czego program nie robi**

Program nie usuwa pozycji oznacza je jedynie jako usunięte (są pomijane przez system) zachowując możliwość odzyskania ich przez administratora. Oznaczenie pozycji jako usuniętej jest realizowane za pomocą ustawienia stanu ilościowego na -1 ( w przypadku stanu magazynowego) na 0 (w przypadku korekt dokumentu sprzedaży) lub przez dodanie na początku nazwiska/nazwy działalności przedrostka „~” (w przypadku kontrahenta).

#### **5. Technologie wykonania**

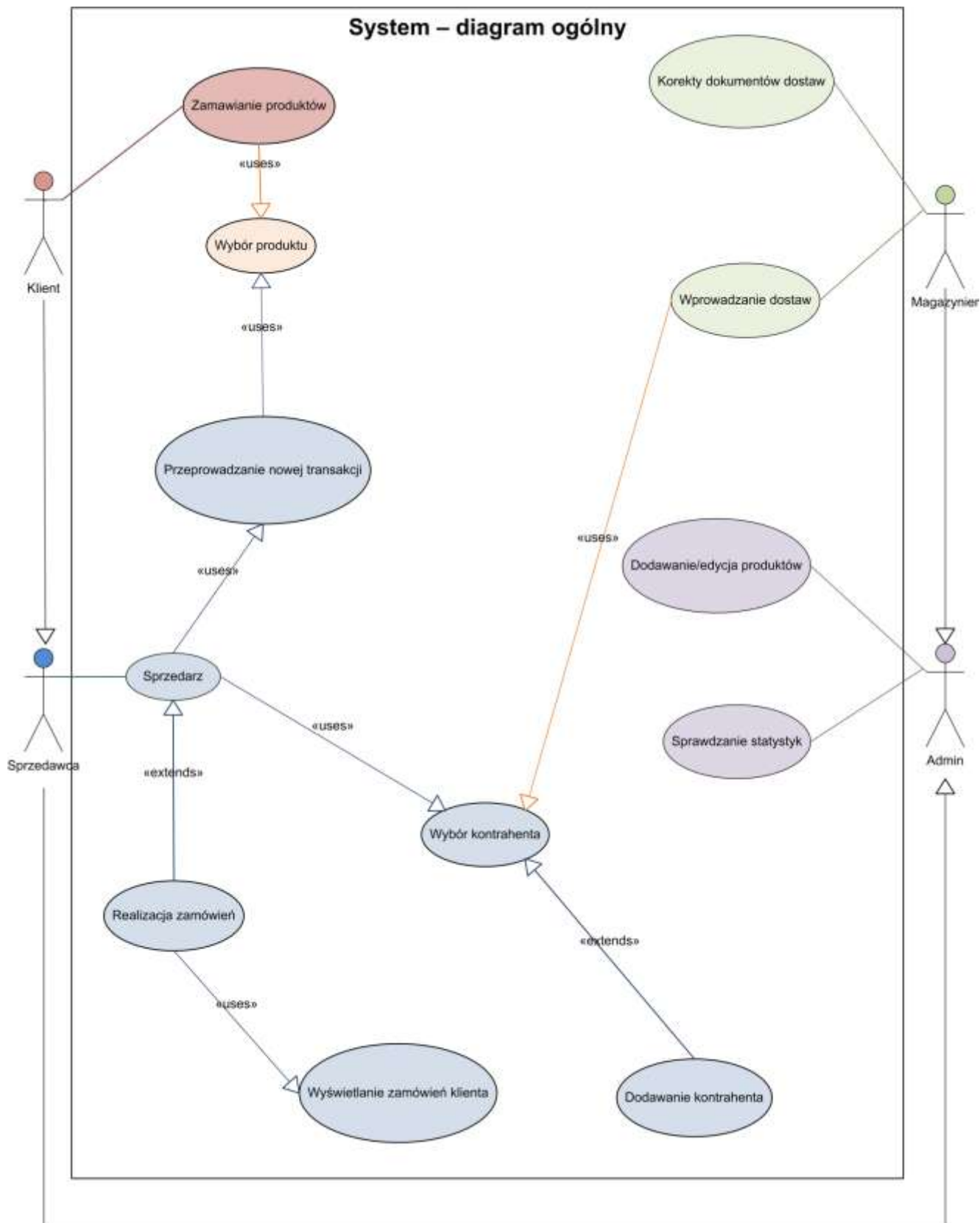
W celu uzyskania niskim nakładem pracy systemu działającego na wielu platformach (Multi-platformowość) program obsługujący sprzedaż zostanie wykonany w technologii .NET.

Językiem wybranym do napisania tej aplikacji został C# ze względu na szybkość i prostotę implementacji tego typu systemu. Natomiast strona internetowa prezentująca ofertę firmy (jest to pozycja opcjonalna) zostanie wykonana w technologii ASP.NET

Baza danych zostanie zaimplementowana w postaci lokalnej bazy danych SQLite(wtedy nie ma możliwości prezentacji oferty na stronie internetowej) lub w wersji zdalnej na serwerze MySQL/ PostgreSQL.

# Diagram przypadków użycia:

## 1. Diagram ogólny:



$$(6*a + 5*b + 7*c + 2*d + 3*e + 4*f + 5*g + 6*h + 7*i) \% 11 - j$$



procesu weryfikacji ma wyświetlić odpowiedni błąd i nie dodać produktu do bazy danych. Gdy weryfikacja przebiegnie pomyślnie wszelkie informacje statystyczne dotyczące produktu jak i jego stan ilościowy są zerowane i następnie wszelkie te dane zostaną zapisane w bazie.

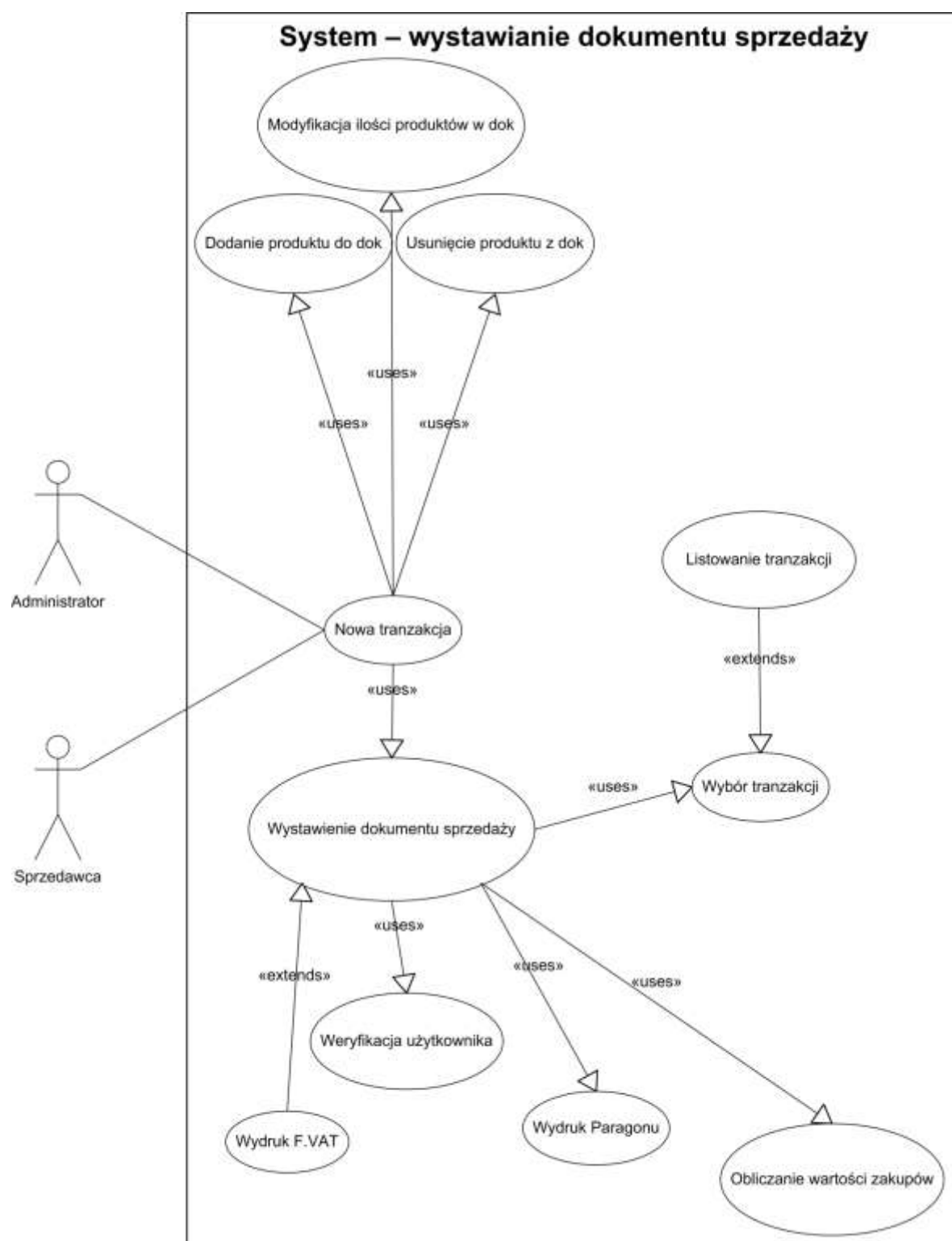
#### 4. Wprowadzanie dostaw:



#### Opis:

Wprowadzanie dostaw zarezerwowane jest dla Magazyniera i Administratora. System ma zweryfikować czy wszystkie pola formularza zostały wypełnione i czy lista towarów jest niepusta. W przypadku braków system ma wyświetlić błąd i nie dodać dostawy do bazy.

## 5. Prowadzenie transakcji i wystawienie dokumentu sprzedaży



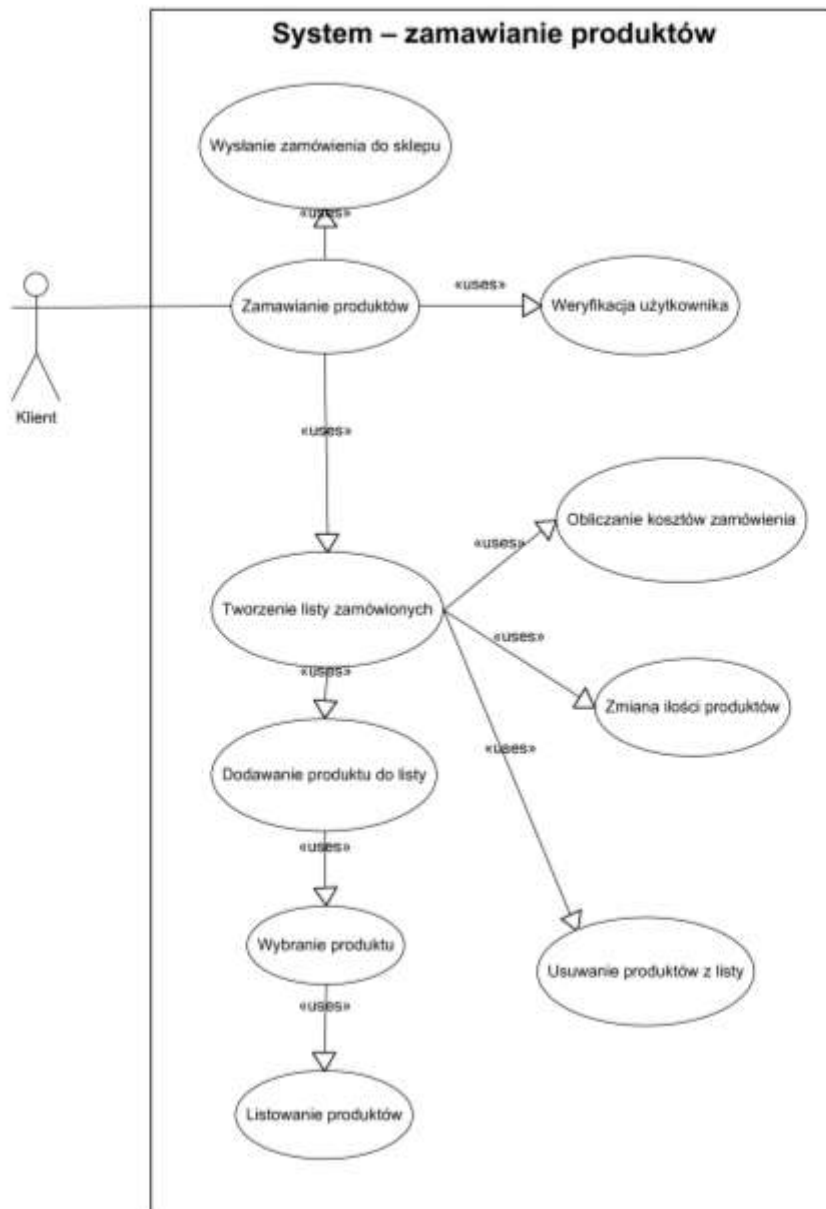
### Opis:

Prowadzenie transakcji i wystawianie dokumentu sprzedaży zarezerwowane jest dla Sprzedawcy i Administratora. Jest to nierozłączny element prowadzenia sprzedaży. Każdy zakup przez klienta musi zakończyć się wystawieniem dokumentu sprzedaży – paragonu. Opcjonalne natomiast jest wystawienie dokumentu zwanego fakturą. Sprzedawca lub



administrator modyfikuje ilość odpowiednich produktów, produkty są dodawane za pomocą wczytywania kodu EAN produktu (możliwe jest też dodawanie produktów za pomocą wyszukiwania/listowania dostępnych produktów). System ma zweryfikować czy lista towarów transakcji nie jest pusta. Jeżeli wszystko jest w porządku system zapisuje nową transakcję w bazie zmniejszając stan ilościowy zakupionych towarów w magazynie i przy pomocy drukarki fiskalnej dokonuje wydruku paragonu i opcjonalnie faktury.

## 6. Zamawianie produktu:

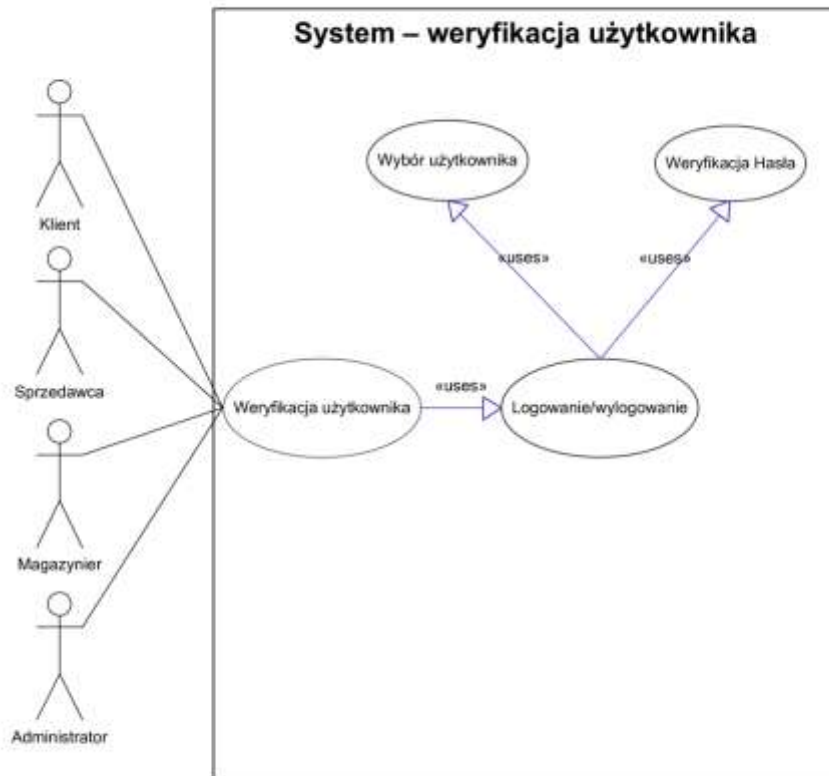


### Opis:

Klient za pośrednictwem strony internetowej po pomyślnym procesie autoryzacji może dokonać zamówienia towaru. Zanim jednak zamówienie zostanie wysłane system sprawdzi

czy lista produktów wybranych do zamówienia jest nie pusta oraz czy wszystkie produkty widnieją w bazie danych jako dostępne w wymaganej ilości(podczas listowania produktów system wyświetla pozycje które są dostępne, ale może się zdarzyć sytuacja że zanim dany klient wyśle zamówienie ktoś inny go uprzedzi i dokona zakupu i stan towaru dostępnego zmniejszy się na tyle że stan ilościowy będzie niższy niż ilość zamawiana przez klienta). W przypadku problemów zostanie wyświetlony odpowiedni komunikat i zamówienie nie zostanie wysłane.

## **7. Weryfikacja użytkownika:**



### **Opis:**

Weryfikacji musi się poddać każdy użytkownik który chce uzyskać dostęp do innych funkcji systemu niż przeglądania oferty za pośrednictwem strony internetowej. Do tego celu potrzebny jest login i hasło. Każdy zarejestrowany klient musi podać swój adres email który jest zarazem jego loginem, na adres ten system automatycznie wysyła losowo wygenerowane hasło. Użytkownicy z większymi uprawnieniami(sprzedawca / magazynier) dostają hasło od administratora. Po podaniu loginu i hasła system sprawdza ich poprawność i czy aktualnie nie są w użyciu (sprawdzi czy nie ma 2 sesji z tym samym użytkownikiem).Jeżeli wszystko pójdzie bezproblemowo użytkownik dostaje dostęp do systemu w zależności od miejsca logowania (strona WWW daje ograniczoną funkcjonalność – jedynie przeglądanie oferty i zamawianie) i od uprawnień użytkownika.

## **8. Pozostałe funkcjonalności:**

- **Usuwanie kontrahenta**

Program ma pytać o potwierdzenie i w przypadku akceptacji oznaczać podanego kontrahenta jako pomijanego w systemie przez dodanie na początku nazwiska znaku '~' (system nie usuwa żadnych danych gdyż są one potrzebne do zachowania ciągłości danych transakcji i do przeglądania historii).

- **Przeglądanie i modyfikacja kontrahentów**

Program ma wyświetlać dane aktualnego rekordu. W odpowiednich polach będzie można modyfikować Imię, Nazwisko, Nazwę działalności, NIP, PESEL, Typ osoby, Dane teleadresowe i rabat specjalny. W przypadku modyfikacji danych program ma weryfikować dane jak w przypadku wstawiania nowego kontrahenta, przy czym nie ma on wstawiać nowego kontrahenta – niezmienione zostają transakcje niezrealizowane, łączna suma wydatków i ilość przeprowadzonych transakcji.

- **Korekta dostawy**

Pola ilości zamówionych towarów w tabeli towarów o określonych ID zmniejszają się o ilość podaną w korekcie. Sam dokument dostawy nie ulega zmianie, natomiast dodawany jest dodatkowy dokument zapisujący różnice.

- **Przeglądanie dostaw**

Przedstawione w formularzu dostaw, widać tam ilość i nazwę zamówionego produktu, jak również podstawowe dane dostawcy i wartość całej dostawy.

- **Przeglądanie transakcji**

Mamy formularz zawierający tytuł i wartość obecnie przeglądanej transakcji, jeśli istnieje kontrahent, którego ID jest podane, to widzimy jego podstawowe dane. ID transakcji nie podlega modyfikacji, data może ulec zmianie.

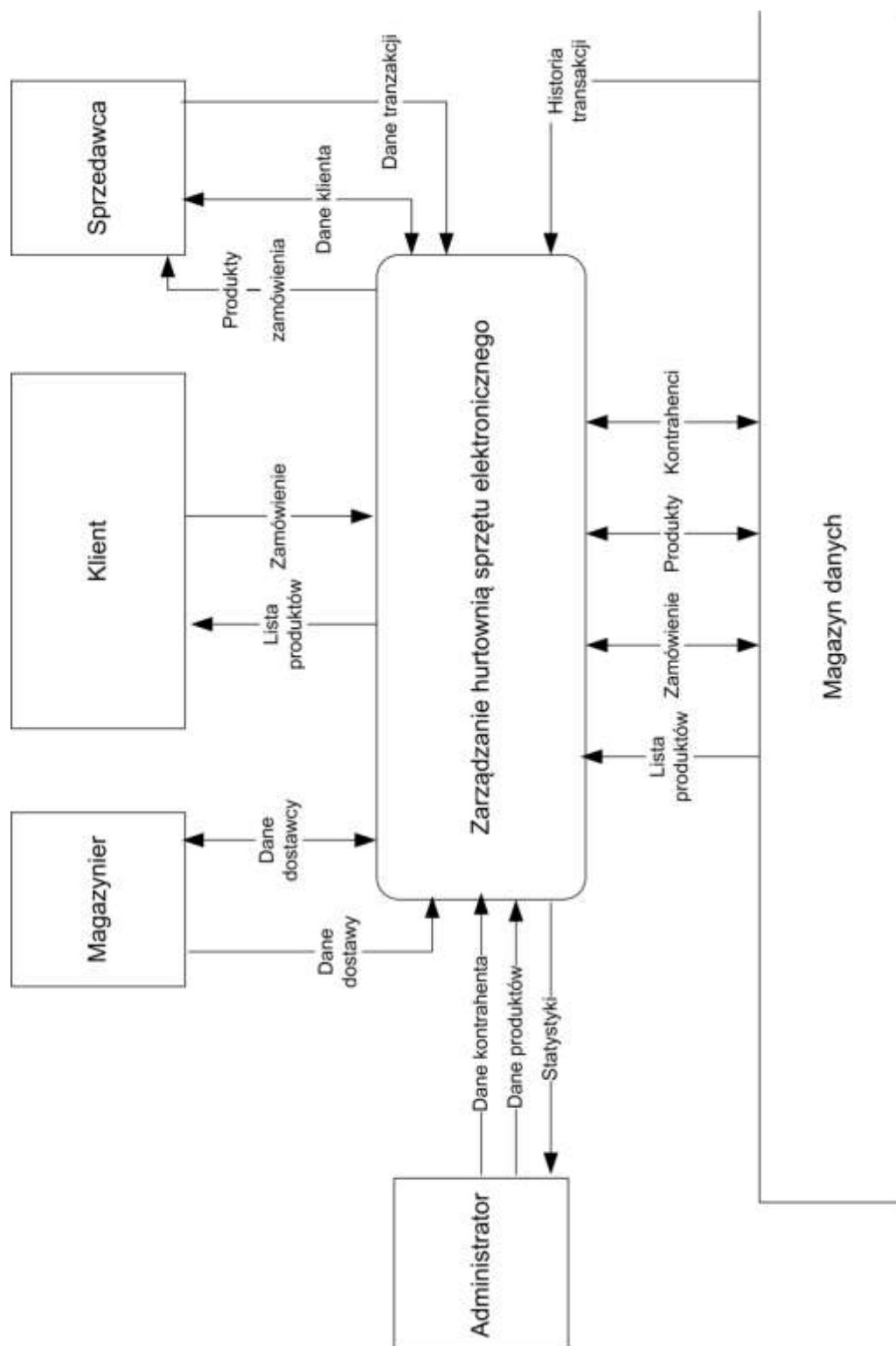
- **Usuwanie towarów**

Operacji usuwania tak naprawdę nie ma – jeśli towaru już nie ma w sprzedaży, to jego ilość ustawia się na -1, co informuje że towar został wycofany z oferty.

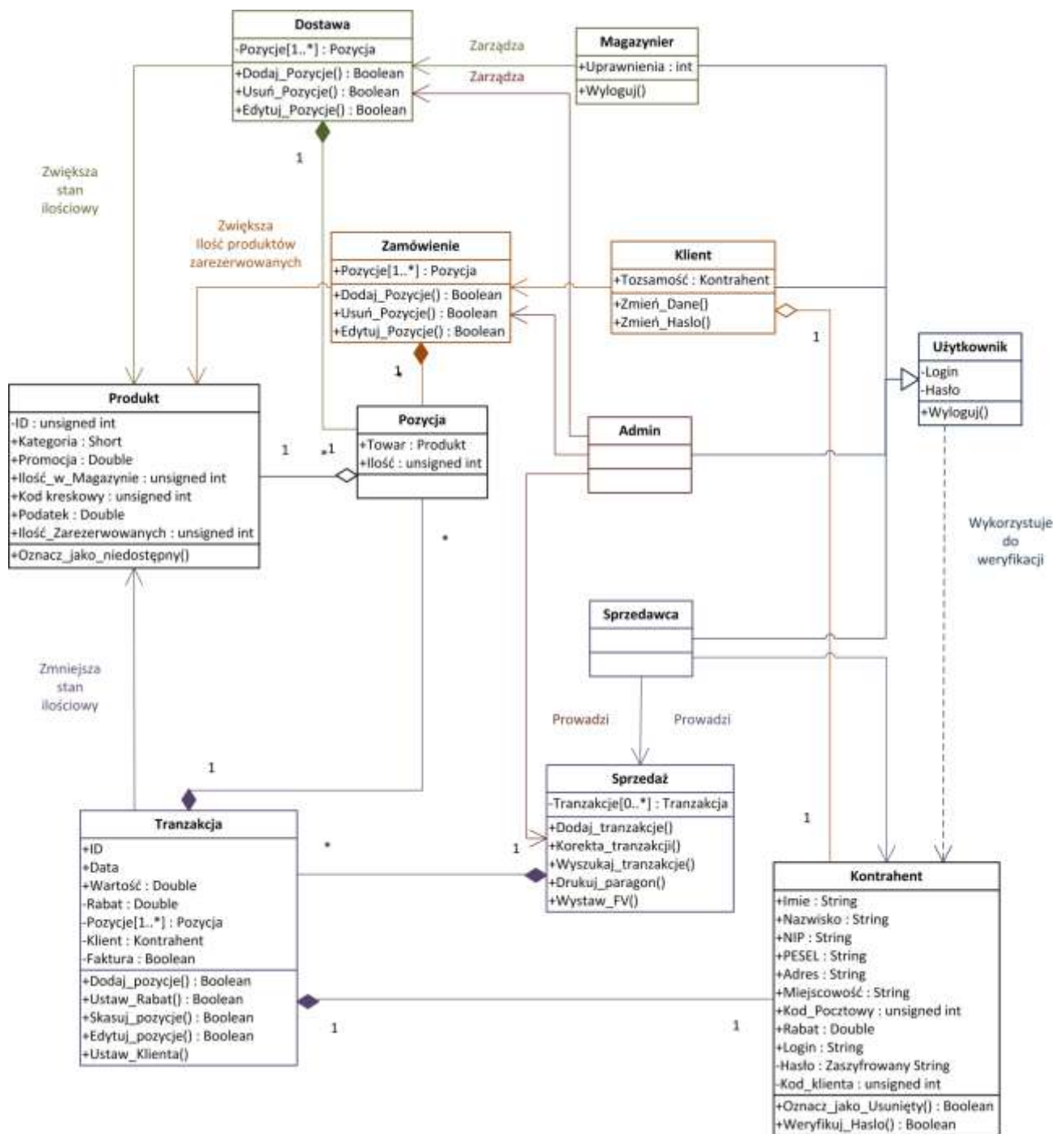
- **Przeglądanie towarów**

W tabeli mamy dane odnośnie produktów, klient widzi tutaj ID produktu, nazwę, cenę, ilość, PKWiU, EAN. Produkty można wyszukiwać (filtrować w oknie) za pomocą wyrażenia lub kategorii.

## Diagram przepływu danych:



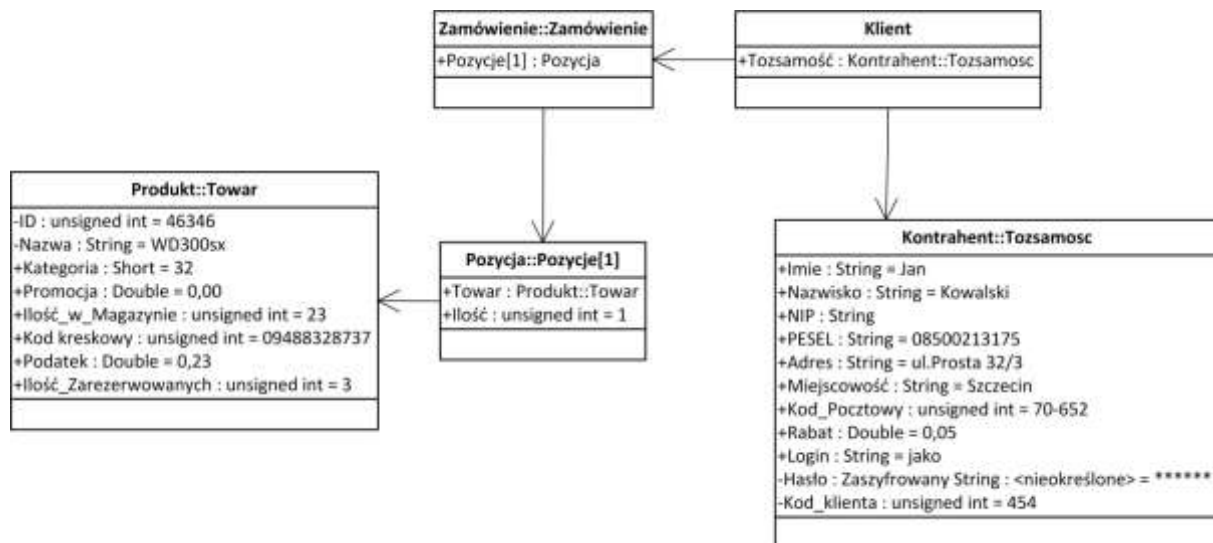
## Diagram klas:



**Opis:**

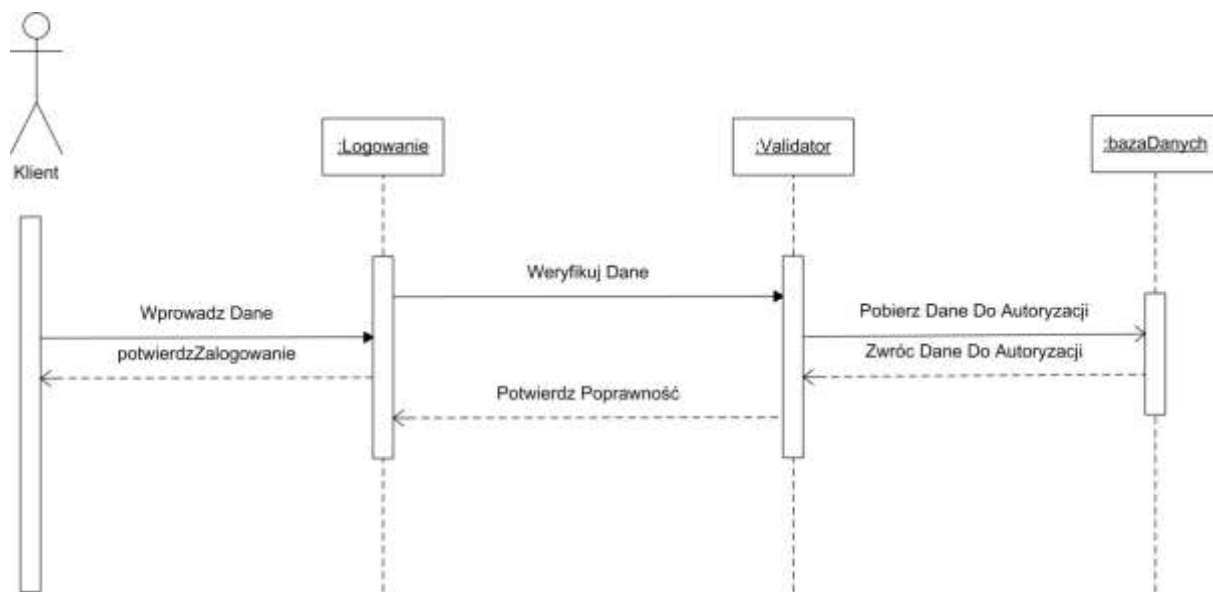


## Diagram obiektów:

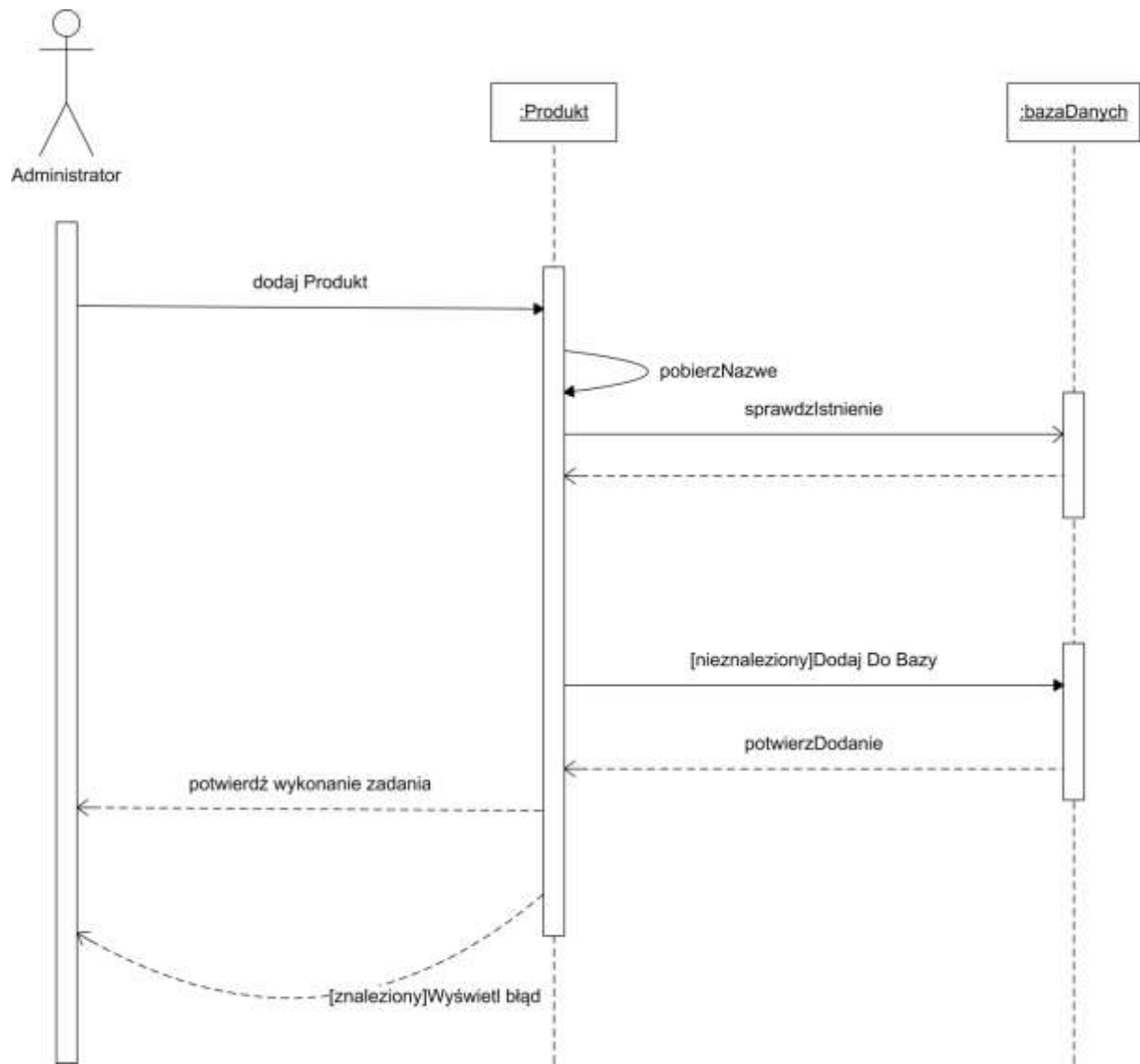


## Diagram Sekwencji:

### 1.Logowanie:



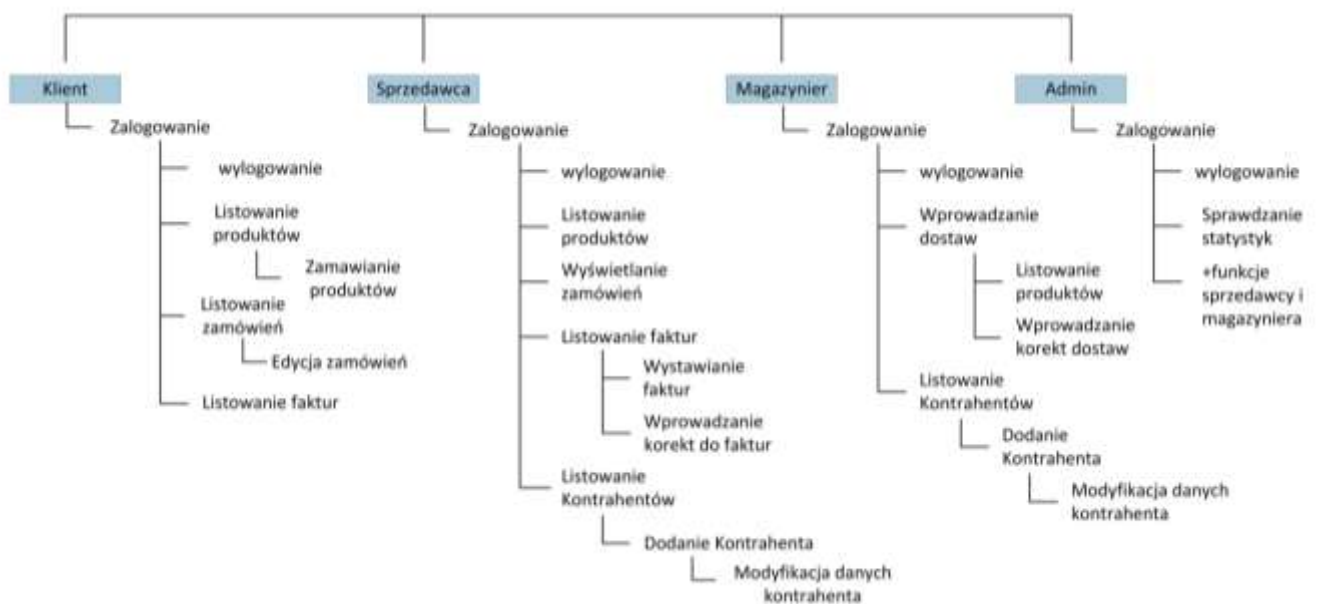
## 2.Dodawanie produktu:



# Hierarchia:



## Hierarchia z podziałem na użytkowników:





# Zabezpieczenia:

Zapewnienie bezpieczeństwa w internetowym systemie będzie realizowane przez następujące funkcje:

- autentyfikację i autoryzację użytkowników w poszczególnych warstwach systemu,
- przesył danych oraz wrażliwych informacji bezpiecznymi kanałami systemu (szyfrowane połączenie),
- unikanie w implementacji aplikacji rozwiązań wrażliwych na atak,
- szyfrowanie danych
- Zabezpieczenie fizycznego dostępu do serwera

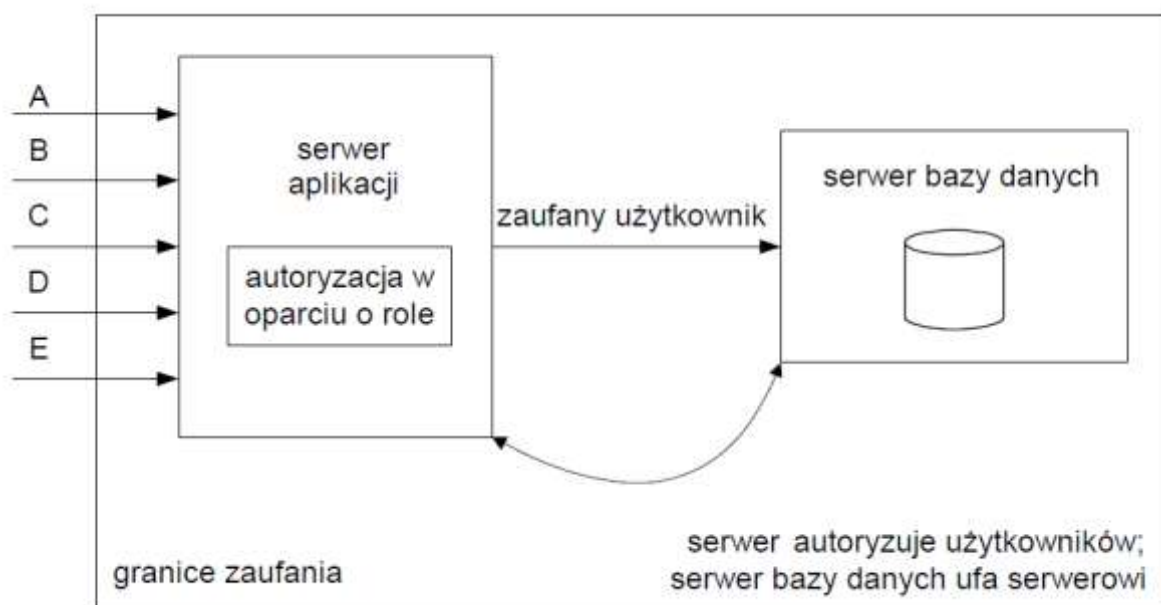
Zastosowany zostanie model zaufanego podsystemu w którym autoryzacja użytkownika jest przeprowadzana w warstwie pośredniej – serwer autoryzacji, tworzącej pewnego rodzaju zaufany podsystem. Serwer bazy danych nie widzi wtedy indywidualnych użytkowników, realizuje tylko polecenia podsystemu autoryzującego, przedstawiającego się ustalonym identyfikatorem (rys. 4). Identyfikator taki ma przypisaną w bazie danych określoną rolę, związaną z pewnym zbiorem uprawnień. Autentyfikacja i autoryzacja użytkowników będzie wykorzystywała technologie SSL. SSL jest protokołem typu klient-serwer pozwalającym na nawiązanie bezpiecznego połączenia z użyciem certyfikatów. Jest on zorientowany głównie na autentyfikację serwera, ale przewiduje również możliwość autoryzacji klienta. Sam algorytm utworzenia połączenia wygląda następująco:

- Klient wysyła do serwera zgłoszenie zawierające obsługiwaną wersję protokołu SSL, dozwolone sposoby szyfrowania i kompresji danych oraz identyfikator sesji. Komunikat ten zawiera również liczbę losową używaną potem przy generowaniu kluczy.
- Serwer odpowiada podobnym komunikatem w którym zwraca klientowi wybrane parametry połączenia: wersję protokołu SSL, rodzaj szyfrowania i kompresji, oraz podobną liczbę losową.
- Serwer wysyła swój certyfikat pozwalając klientowi na sprawdzenie swojej tożsamości
- Po przesłaniu swojego certyfikatu serwer zawiadamia, że chciałby otrzymać certyfikat klienta
- Po otrzymaniu komunikatu ServerHelloDone klient odsyła swój certyfikat (certyfikat jest dawany każdemu użytkownikowi systemu i jest inny dla każdej grupy-klient, sprzedawca, admin czy magazynier lub dla każdego użytkownika jest przydzielony inny lecz niezalecane bo dodawanie nowego klienta będzie wymagało wygenerowania klucza i zwiększonych wysiłków administratora)
- Klient musi potwierdzić, że faktycznie posiada klucz prywatny odpowiadający wysłanemu certyfikatowi. W tym celu klient podpisuje swoim kluczem prywatnym skrót wszystkich dotychczas ustalonych danych o połączeniu i wysyła go korzystając z tego komunikatu.
- Serwer wysyła informację o swoim kluczu publicznym. Rodzaj i długość tego klucza jest określony przez typ algorytmu przesłany w poprzednim komunikacie.
- Serwer zawiadamia, że klient może przejść do następnej fazy zestawiania połączenia.
- Klient na podstawie ustalonych w poprzednich komunikatach dwóch liczb losowych (swojej i serwera) generuje klucz sesji używany do faktycznej wymiany danych. Następnie wysyła go serwerowi używając jego klucza publicznego. Wygenerowany klucz jest kluczem algorytmu symetrycznego w naszym przypadku będzie to AES (algorytm ten został wybrany w związku z zapewnianym przez niego

bezpieczeństwem, np. w porównaniu z DES-em, w którym niewielka długość klucza nie zapewnia odpowiedniego poziomu bezpieczeństwa).

- Klient zawiadamia, że serwer może przełączyć się na komunikację szyfrowaną, że jest gotowy do odbierania danych zakodowanych.
- Serwer zawiadamia, że wykonał polecenie - od tej pory wysyłał będzie tylko zaszyfrowane informacje.
- Następuje dodatkowa autoryzacja z wykorzystaniem login i hasła

Implementacja powyższego rozwiązania nie będzie stanowiła większego problemu gdyż w przypadku programu-klienta platforma .Net zawiera klasę SslStream w System.Net.Security która zawiera wszystkie wymagane przez nas funkcje i jest dostępna od wersji 3.5, natomiast serwer będzie wykorzystywał popularny i prosty w obsłudze pakiet openssl.

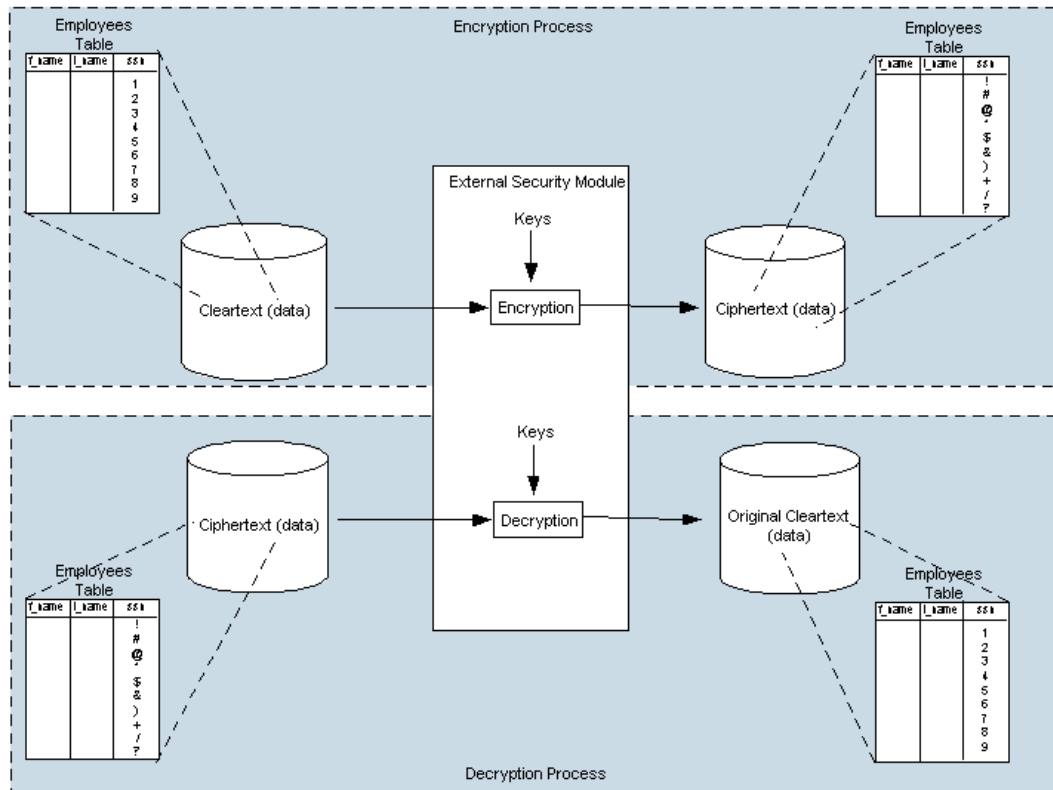


Zostanie również zastosowany mechanizm perspektyw pozwalający tworzyć wirtualne tablice udostępniające użytkownikowi fragmenty zasobów jednej lub wielu tablic rzeczywistych, w formie prostej lub przetworzonej. Przy wykorzystaniu tego mechanizmu mogą być przyznawane użytkownikom prawa dostępu nie do tablic rzeczywistych, a tylko do perspektyw. Pozwala to na wprowadzenie różnorodnych ograniczeń, w tym np. udostępnienie tylko wybranych kolumn bądź też wybranych wierszy tablic.

Również dane na serwerze bazodanowym będą szyfrowane. Dane te przechowywane na serwerze w postaci jawnej mogą się bowiem stać dość interesującym łupem dla włamywaczy. Do zaszyfrowania danych będzie można użyć AES, lub 3DES, przy pomocy mechanizmu TDE.

Mechanizm TDE przeznaczony jest do szyfrowania kolumn tabeli w celu podwyższenia stopnia poufności danych i kontroli integralności danych powszechnie stosowany między innymi w rozwiązaniach bazodanowych Oracle czy Microsoftu. Transparent Data Encryption obsługuje szyfrowanie kolumn tabeli przy użyciu algorytmów kryptograficznych 3DES lub AES z kluczem 128-bitowym, 192-bitowym lub 256-bitowym. Do kontroli integralności danych wykorzystywana jest kryptograficzna funkcja skrótu SHA-1 generująca skrót o

długości 160-bitów.



AES wykonuje 10 (klucz 128 bitów), 12 (klucz 192 bity) lub 14 (klucz 256 bitów) rund szyfrujących. Składają się one z substytucji wstępnej, permutacji macierzowej (mieszanie wierszy, mieszanie kolumn) i modyfikacji za pomocą klucza. Funkcja substytucyjna ma bardzo oryginalną konstrukcję, która uodparnia ten algorytm na znane ataki kryptoanalizy różnicowej i liniowej.

3DES używa takich samych rozmiarów bloków oraz trybów jak zwykły DES. 3DES z trzema różnymi kluczami (3TDES) ma siłę 168 bitów (3x56-bitowy).

Pomimo, że TDE nie jest zaimplementowane jako funkcja w żadnym przez nas wykorzystywanym systemie bazodanowym PostgreSQL czy MySQL, to skorzystamy z darmowego rozwiązania realizujące podobne funkcje - eCryptfs.

# Diagramy stanów:

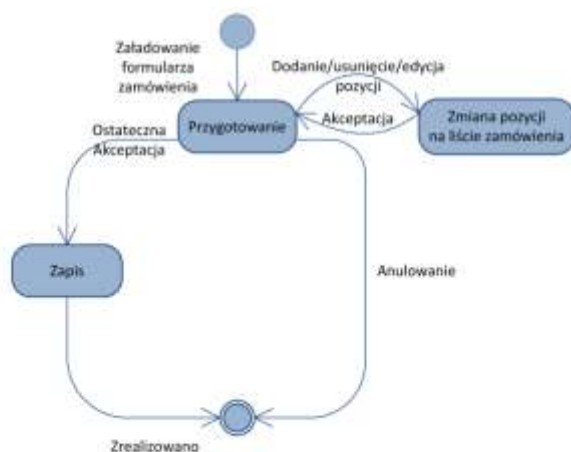
## Dostawa:



## Logowanie:



## Złożenie zamówienia:



# Interfejs graficzny:

**Logowanie**

Login:

Hasło:

**Sprzedaż**

**Sprzedaż produktów**

Kontrahent:



☒ Faktura Razem: 0.00 zł (netto)  
Razem: 0.00 zł (brutto)

**Dokumenty handlowe**

**Dokumenty handlowe**

Sprzedaż				
Faktur sprzedaży				
Paragony				
Raporty				
Korekty				
Zakup				
Faktury zakupu				

**Produkty**

**Lista produktów**

Hub USB				
Komputery				
Stacjonarne				
Laptopy				
Notbooki				
Tablety				
Komponenty komp.				
Procesory				

**Kontrahenci**

**Kartoteka kontrahentów**

Dostawcy				
Odbiorcy				
Dostawcy i Odbiorcy				

**Dodanie/Edycja Produktu**

Nazwa:  Oryginalna nazwa:

Typ:

Kategoria:  Kod kreskowy:

Index:

Stawka VAT:  Jednostka magazynowa:

Zakup: Sprzedawcy:  Cena zakupu (netto):

SWW/KU:

PKW/U:

Cena sprzedaży (netto):

**Dodawanie/Edycja Kontrahenta**

Nazwa:  Kraj:

Typ:  Kod Pocztowy:

Nip:  Miejscowość:

☒ Płatnik VAT Ulica/Lokal:

Regon:  Województwo:

Kontakty:

Imię	Nazwisko	Adres	Tel.

**Dokumenty magazynowe**

**Dokumenty Magazynowe**

Przychód				
Przyjęcie zewnętrzne				
Usługi				
Korekta				
Rozchód				
Wydanie zewnętrzne				
Usługi				
Korekta				

**Dodawanie/Edycja Dokumentu Magazynowego**

Typ dokumentu:  Kontrahent:

Numer:  Z dnia:



Razem: 0.00 zł (netto)

**Zamówienia**

Zrealizowanie				
Częściowo zrealizowane				
Niezrealizowane				
Anulowane				

**Menu**

Zalogowano jako: Sprzedawca1

**Dodaj/Edytuj Zamówienie**

Numer:  Kontrahent:

Termin realizacji:  Z dnia:

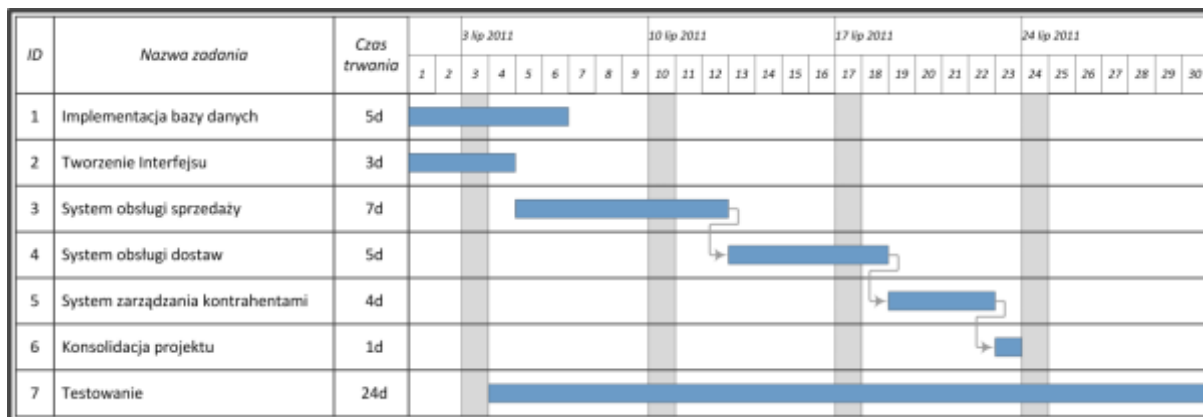


Razem: 0.00 zł (netto)

**Statystyki**

Razem: 0.00 zł (netto)  
Razem: 0.00 zł (brutto)

## Diagram Gantta:



Czas potrzebny do zrealizowania poszczególnych zadań został obliczony przy pomocy delfickiej zgodnie ze wzorem:

$$\text{Oszacowanie} = (P + 4A + O) / 6$$

Gdzie:

P – ocena pesymistyczna

A – ocena średnia

O – ocena optymistyczna

W metodzie tej wykorzystano opinie 3 ekspertów ( studentów informatyki), a oszacowany wynik został zaokrąglony do pełnych dni.

Zespół realizujący te zadanie będzie się składał z jednego programisty , projektanta GUI, specjalisty od baz danych i tester. Nad wszystkim dodatkowo piecze będzie miał kierownik projektu który będzie koordynował prace.



# Kosztorys:

## 1.Koszty pracy:

Id.	Specjalista	Koszt wymaganego sprzętu	Koszt wymaganego oprogramowania	Koszt zatrudnienia	Suma
1.	Programista c#	-2499zł (komputer PC )	-3070zł (Visual Studio) -420zł (Windows 7)	25,00zł * 136h	6890zł
2.	Kierownik projektu	-1999zł (komputer PC)	-0zł (Ubuntu) -0zł (LibreOffice)	27,00zł * 208h	5616zł
3.	Tester	-2499zł (komputer PC)	-420zł (Windows 7)	18,50zł * 192h	3972zł
4.	Specjalista od baz danych	-2499zł (komputer PC)	-0zł (Ubuntu) -0zł ( PostgreSQL 9)	23,50zł * 40h	940zł
5.	Projektant GUI	Korzysta ze sprzętu i oprogramowania z poz. 1		18,00zł * 24h	432zł

## 2.Koszty dodatkowe:

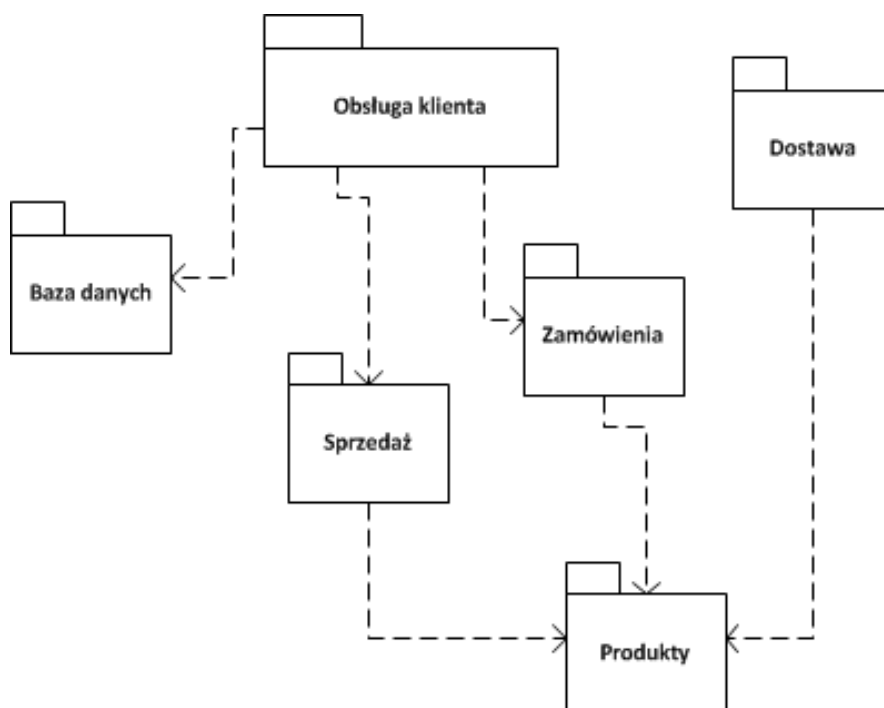
Id.	Pozycja	Koszt
1	Wynajem lokalu 23m <sup>2</sup> (50zł/metr)	1250zł
2	Media	99zł
3	Prąd	70zł
4	Inne (kawa, art. biurowe)	50zł

## 3.Podsumowanie:

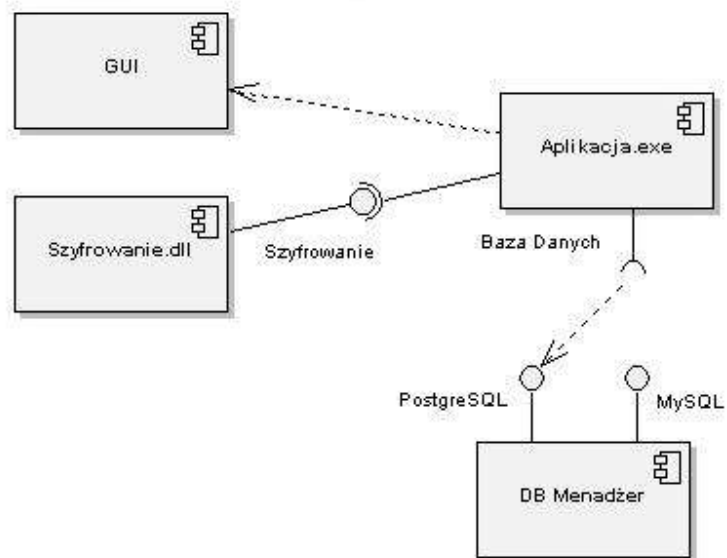
Id.	Pozycja	Koszt
1	Zatrudnienia	13 940zł
2	Sprzętu	9 496zł
3	Oprogramowania	3 910zł
4	Dodatkowe	1 469zł
-	<b>Suma:</b>	<b>17 387zł</b>



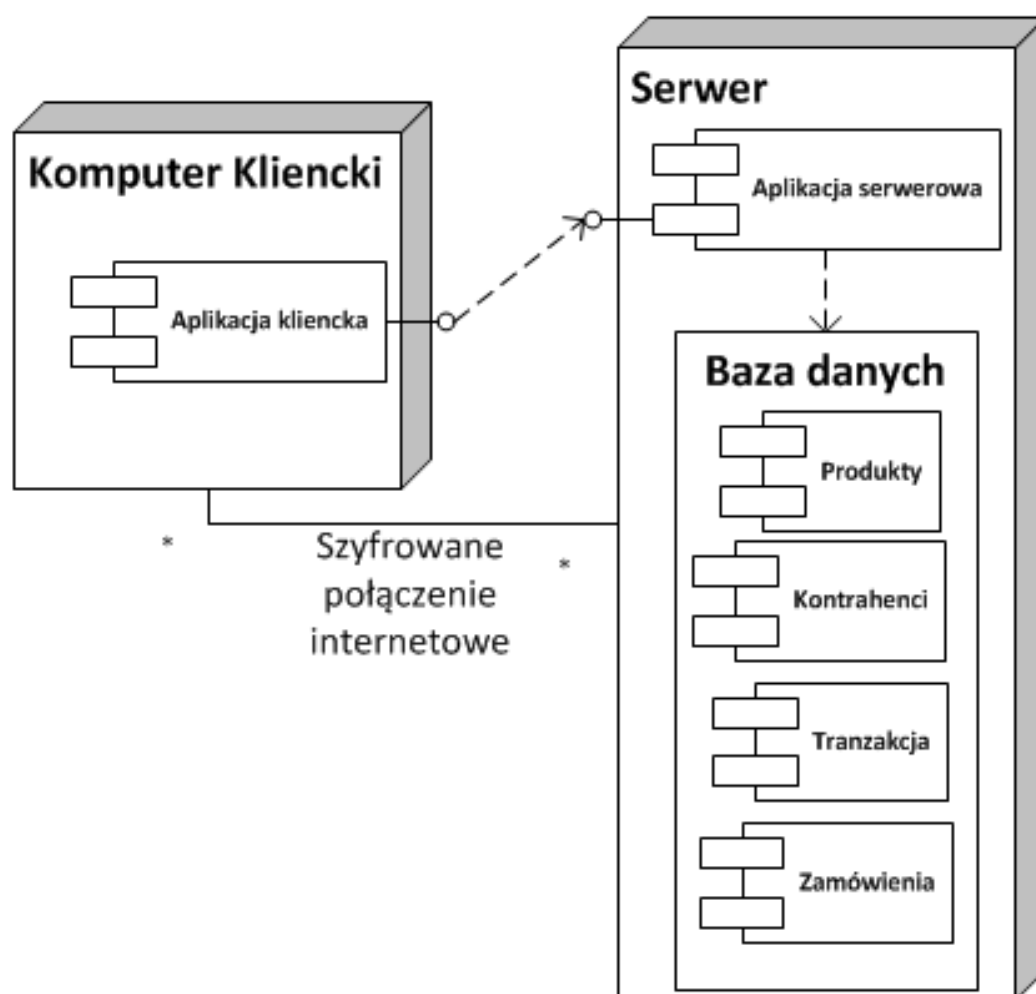
## Diagram pakietów:



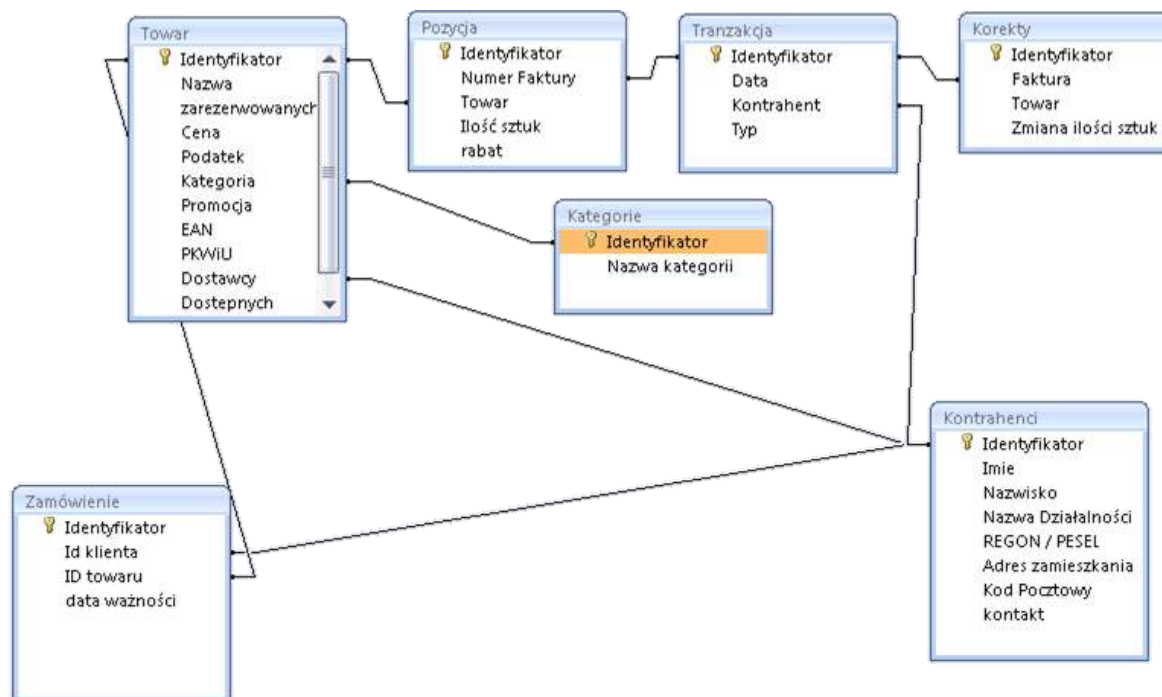
## Diagram komponentów:



## Diagram wdrożenia:



# Baza Danych:



## 1.Dane przechowywane przez system

### ➤ Zarządzanie stanem towarów w poszczególnych magazynach należących do hurtowni – będzie się tu znajdował cały asortyment hurtowni

- ID Produktu – unikalna wartość
- Nazwa Produktu – 255 znaków UTF-8
- Ilość w sprzedaży/zamówionych/w magazynie/zarezerwowanych – Cztery pola liczb szesnastobitowych ze znakiem
- Cena – Liczba zmiennoprzecinkowa, dokładnie dwa miejsca po przecinku
- Podatek od produktu – podatek od wartości dodanej (procentowy) jako procent od ceny ilości produktu przechowywany w liczbie zmiennoprzecinkowej posiadającej dwa miejsca po przecinku
- Kategoria – wyrażona liczbą całkowitą bez znaku
- Promocja – procentowa wartość upustu ceny produktu
- EAN – 13 cyfr dziesiętnych oznaczających kod kreskowy
- PKWiU – 8 cyfr dziesiętnych określających produkt

### ➤ Zarządzanie kontrahentami

- Informacje o kontrahentach – tutaj będą się znajdowali wszyscy kontrahenci
  - ID kontrahenta – unikalna wartość (nadawana losowo lub sekwencyjnie)
  - Imię – ciąg 50 znaków UTF-8 (może być pusty dla typu osoby: działalność gospodarcza)
  - Nazwisko – ciąg 255 znaków UTF-8 (może być pusty dla typu osoby: działalność gospodarcza)
  - Nazwa działalności – ciąg 255 znaków UTF-8 (może być pusty dla typu osoby: osoba fizyczna)
  - NIP – numer identyfikacji podatkowej (dla działalności gospodarczej)

- PESEL – ciąg liczb jednoznacznie określający osobę (dla osoby fizycznej)
- Typ osoby (działalność gospodarcza/osoba fizyczna) – jeden bit oznaczający typ osoby
- Dane teleadresowe
  - Adres zamieszkania / siedziby firmy (Ulica, nr domu i mieszkania) – ciąg 1024 znaków UTF-8
  - Kod pocztowy – 5 cyfr dziesiętnych
  - Miejscowość – ciąg 255 znaków UTF-8
  - Numer kontaktowy – Do 10 cyfr dziesiętnych
  - Adres e-mail – Do 255 znaków UTF-8
  - Numer konta bankowego – 26 cyfr dziesiętnych
- Ilość przeprowadzonych transakcji – do 5 cyfr dziesiętnych
- łączna suma wydatków – wartość liczbowa do 9 cyfr dziesiętnych
- Rabat specjalny – procentowy rabat na wszystkie kategorie produktów zapisywany liczbą zmiennoprzecinkową
- Transakcje niezrealizowane (ustalone) – partie towarów zamówione przez kontrahenta, ale nie odebrane, lista Numerów zamówień oddzielona przecinkami

#### ➤ Zarządzanie dostawcami/dostawami

- Zarządzanie dostawcami
  - ID Kontrahenta będącego dostawcą – musi odnosić się do istniejącego kontrahenta w bazie
  - Oferta produktowa dostawcy – ciąg znaków ID produktów oferowanych przez dostawcę oddzielonych przecinkiem
- Dostawy – dostawy, które nie zasiły magazynu
  - ID dostawy – unikalny ID nadawany każdej dostawie
  - ID kontrahenta – powiązane z istniejącym kontrahentem
  - Wartość dostawy – wyrażona liczbą zmiennoprzecinkową wartość dostawy w PLN
- Szczegóły dostaw – dokładny spis produktów w dostawie
  - ID dostawy – unikalne ID pochodzące z tabeli dostaw
  - ID Produktu – ID pochodzące z tabeli produktów
  - Ilość – 4 cyfry dziesiętne

#### ➤ Zarządzanie transakcjami

- Transakcje – dochody i rozchody uzyskane w transakcjach w ramach prowadzonej działalności gospodarczej
  - ID transakcji – Unikalna wartość nadana transakcji
  - Data transakcji – UNIX TIMESTAMP
  - Wartość – zmiennoprzecinkowa liczba
  - Tytuł (opcjonalne) – ciąg 255 znaków UTF-8
  - ID kontrahenta (opcjonalne) – ID kontrahenta objętego transakcją (jeśli dotyczy)
- Wystawione faktury
  - Numer faktury – Unikalny (zgodny z wystawioną fakturą)
  - Data wystawienia – UNIX TIMESTAMP
  - ID Kontrahenta – ID kontrahenta objętego transakcją
  - Wartość netto – zmiennoprzecinkowa liczba
  - Wartość brutto – zmiennoprzecinkowa liczba
- Zamówienia
  - Numer – liczba szesnastobitowa (unikalna dla każdego zamówienia)
  - Data odebrania – UNIX TIMESTAMP