

# Przegląd diagramów UML

Diagramy „analityczne” – wybór diagramów używanych przez analityka biznesowego

## Spis treści

UML i diagramy .....	2
Diagram klas .....	3
Diagram przejść stanów .....	5
Diagram przypadków użycia.....	6
Przypadki użycia .....	8
Specyfikacja przypadków użycia.....	8
Przykłady specyfikacji przypadku użycia .....	9
Diagram aktywności .....	11
Rysunek 1 Przykład diagramu klas dla e-sklepu .....	4
Rysunek 2 Przykład diagramu klas dla systemu bankowego.....	4
Rysunek 3 Przykład diagramu przejść stanów dla obiektu konto klienta (system bankowy) .....	5
Rysunek 4 Przykład diagramu przejść stanów dla obiektu przelew (system bankowy).....	6
Rysunek 5 Przykład diagramu przypadków użycia dla systemu bankowego .....	7
Rysunek 6 Przykład diagramu przypadków użycia dla systemu bankowego - przelewy.....	7
Rysunek 7 Przykład diagramu przypadków użycia dla e-sklepu.....	8
Rysunek 8 Przykład diagramu aktywności - usuwanie konta klienta .....	12
Rysunek 9 Przykład diagramu aktywności wyświetlanie danych profilu klienta.....	14



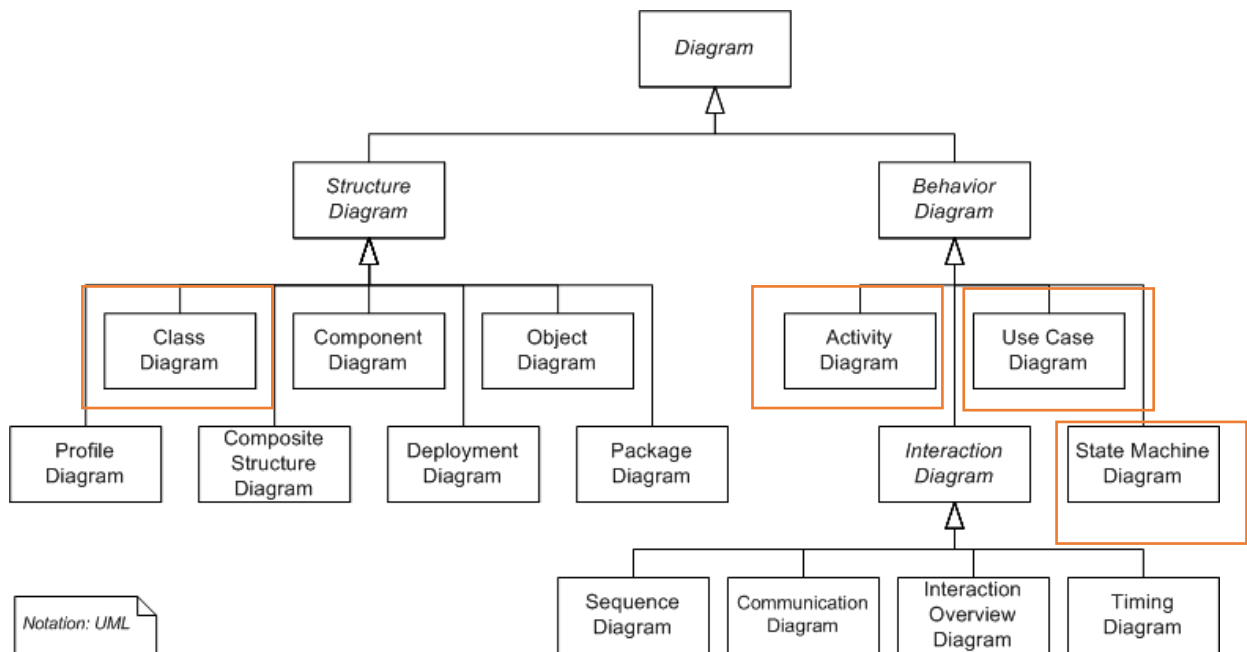
## UML i diagramy

UML (Unified Modeling Language) to standardowy język modelowania, który służy do opisu, projektowania i dokumentowania systemów informatycznych. UML dostarcza zestawu notacji graficznych i reguł semantycznych, które umożliwiają tworzenie spójnych i czytelnych diagramów, które opisują różne aspekty systemu. Oto kilka diagramów, które można używać w UML:

- Diagram klas: Diagram klas służy do modelowania struktury obiektowej systemu, opisując klasy, atrybuty, metody i zależności między nimi.
- Diagram komponentów: Diagram komponentów służy do modelowania struktury systemu na poziomie komponentów i zależności między nimi. Przedstawia komponenty, interfejsy, zależności i strukturę fizyczną systemu.
- Diagram wdrożenia: Diagram wdrożenia opisuje sposób w jaki system lub komponenty systemu są wdrażane na fizycznych środowiskach. Przedstawia węzły (serwery, urządzenia), komponenty wdrażane na tych węzłach i zależności między nimi.
- Diagram przypadków użycia: Diagram przypadków użycia opisuje funkcjonalność systemu z punktu widzenia użytkowników. Przedstawia interakcje między aktorami (użytkownikami systemu) i przypadkami użycia (funkcjonalnościami systemu).
- Diagram sekwencji: Diagram sekwencji pokazuje interakcje między obiektami w czasie, przedstawiając sekwencję komunikacji między nimi. Przedstawia zarówno obiekty, jak i komunikaty, które są przesyłane między nimi.
- Diagram komunikacji: Diagram komunikacji opisuje interakcje między obiektami w określonym kontekście lub scenariuszu. Przedstawia strukturę komunikacji między obiektami, wskazując przepływ sterowania i komunikacji.
- Diagram stanów: Diagram stanów służy do modelowania zachowań obiektów lub systemów w kontekście różnych stanów, w jakich mogą się znajdować. Przedstawia stany, przejścia między nimi oraz akcje wykonywane w poszczególnych stanach.
- Diagram aktywności: Diagram aktywności opisuje sekwencję działań, procesy biznesowe lub algorytmy w systemie. Przedstawia przepływ sterowania, akcje, decyzje, rozgałęzienia i łączenia w celu modelowania różnych scenariuszy działania.

To tylko kilka przykładów diagramów dostępnych w UML. Istnieje wiele innych diagramów, takich jak diagramy interakcji, diagramy pakietów, czy diagramy czasowe.





## Diagram klas

Diagram klas (ang. *class diagram*) w UML (Unified Modeling Language) jest graficznym sposobem przedstawiania definicji i struktury klas, relacji między nimi oraz ich właściwości i metod. Diagram ten służy do modelowania statycznych aspektów systemu.

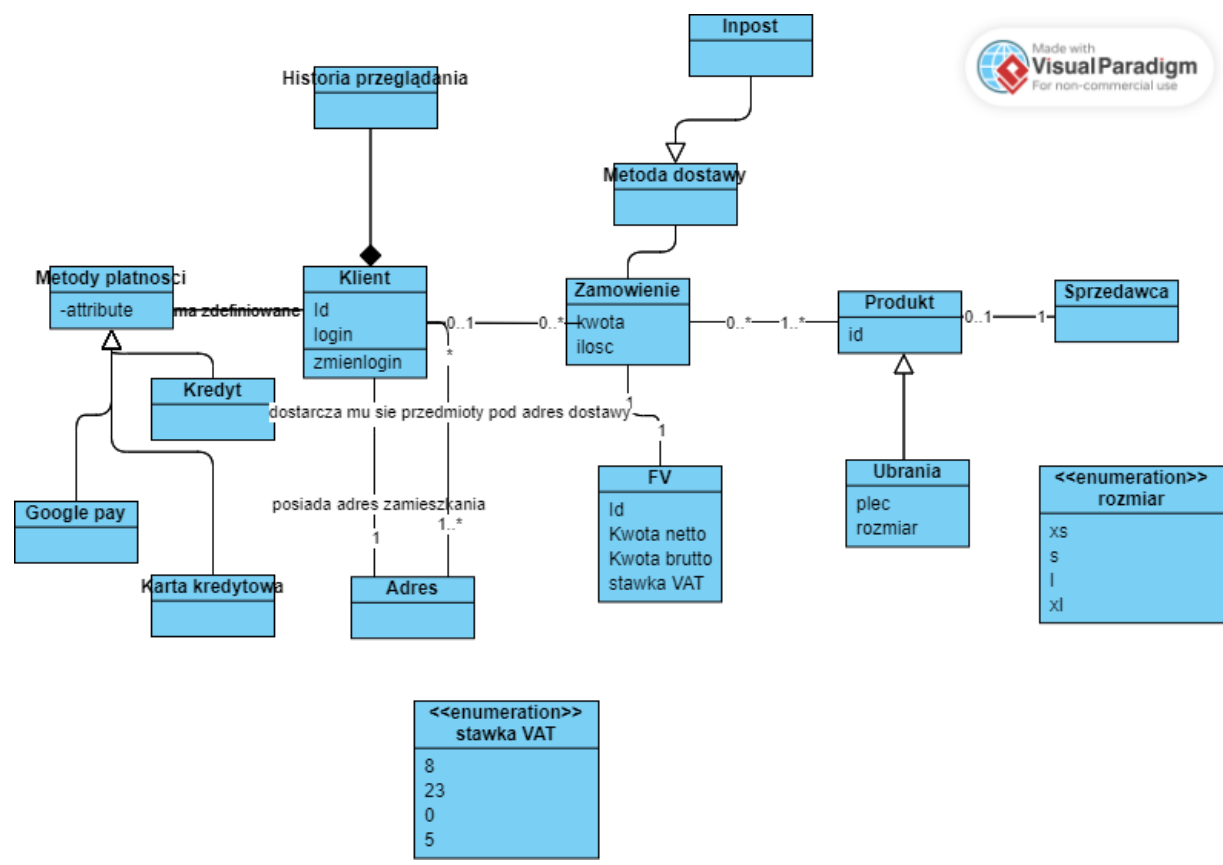
Typowe zastosowania to:

- Przedstawienie klas i ich atrybutów (cech) oraz metod (operacji). Klasa jest reprezentowana jako prostokąt, w którym znajdują się nazwa klasy, atrybuty i metody. Diagram klas pokazuje także relacje między klasami..
- Wyrażenie zależności i relacji między klasami. Relacje to skojarzenia między klasami – powiązanie klas (danych) w pewien sposób (np. klient składa zamówienie). Zależności to relacje, w których jedna klasa zależy od innej klasy, na przykład przez użycie jej metod lub atrybutów. Specyficzną relacją jest kompozycja, zwane relacją całość – część.

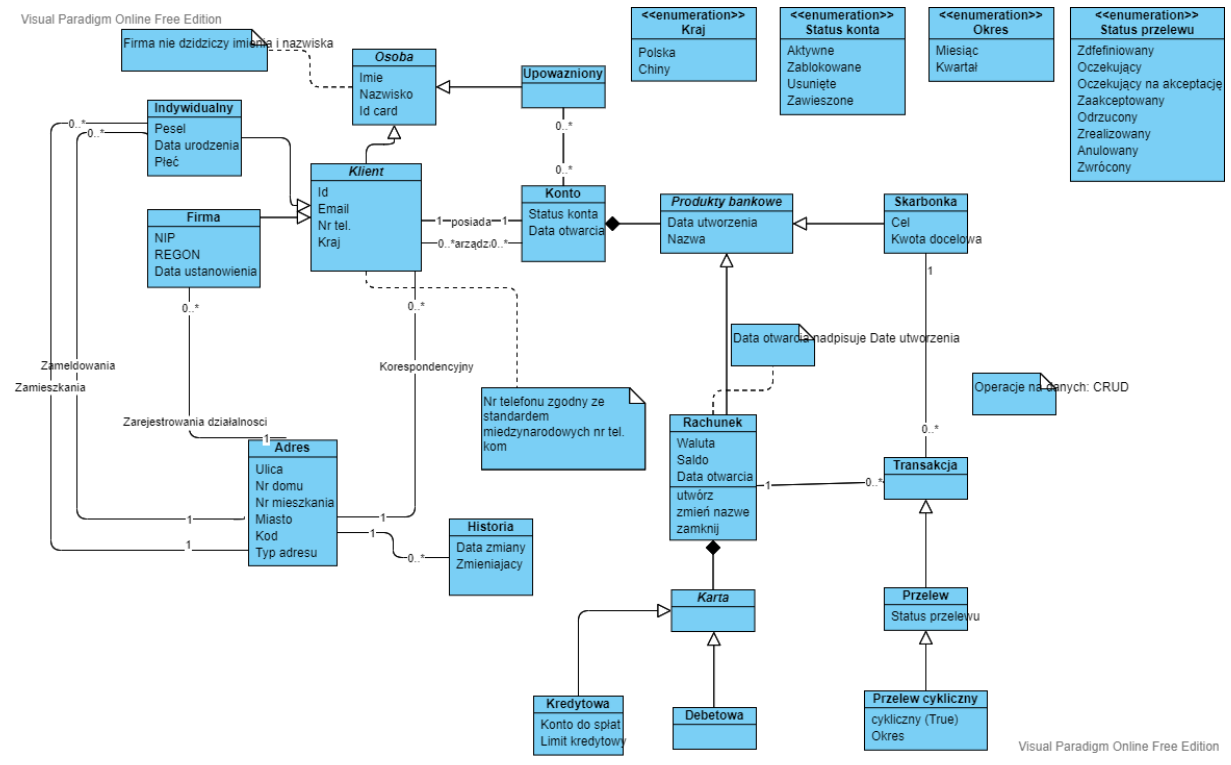
Kluczowe elementy diagramu klas:

- Klasy: Klasy reprezentują typy obiektów w systemie. Są to abstrakcje, które opisują wspólne cechy, atrybuty i zachowania obiektów. Klasy są reprezentowane jako prostokąty, w których umieszcza się nazwę klasy.
  - Atrybuty: Atrybuty są cechami lub właściwościami klasy. Opisują stan obiektów danej klasy. Atrybuty mogą mieć nazwy i typy danych. Są reprezentowane jako nazwa atrybutu wewnątrz prostokąta reprezentującego klasę.
  - Metody: Metody to operacje lub funkcje, które można wykonywać na obiektach danej klasy. Określają zachowanie obiektów danej klasy. Metody mogą mieć nazwy, argumenty, typy zwracane i możliwe parametry. Są reprezentowane jako nazwa metody wewnątrz prostokąta reprezentującego klasę.
- Relacje: Relacje opisują relacje między dwiema klasami. Wskazują, że istnieje pewien związek lub powiązanie między obiektami tych klas. Mogą mieć nazwy, kierunki, kardynalności (np. jeden do jednego, jeden do wielu, wiele do wielu) oraz etykiety, które opisują naturę relacji.





Rysunek 1 Przykład diagramu klas dla e-sklepu



Rysunek 2 Przykład diagramu klas dla systemu bankowego



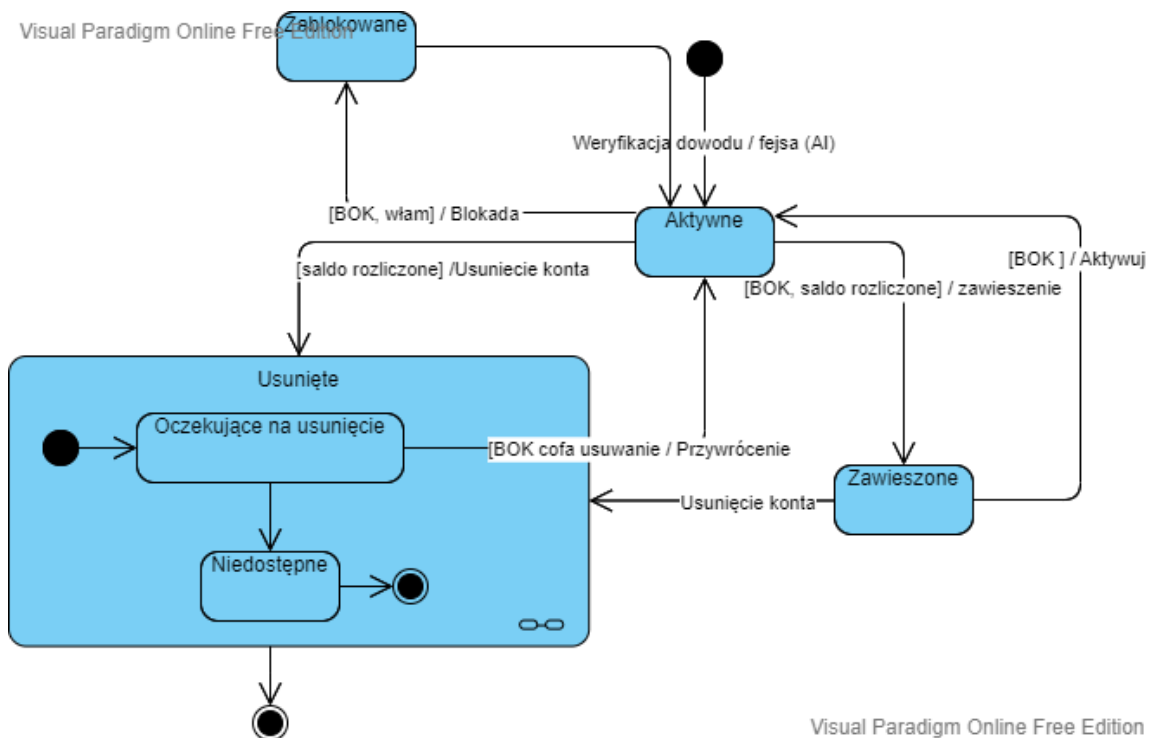
## Diagram przejść stanów

Diagram przejść stanów (ang. *state transition diagram*) w UML (Unified Modeling Language) jest graficznym narzędziem do modelowania zachowania obiektu lub systemu poprzez reprezentację różnych stanów, przejść między nimi oraz zdarzeń, które powodują te przejścia. Diagram ten jest szczególnie przydatny w modelowaniu systemów, które mają dynamiczne zachowanie i przejmują różne stany w zależności od zdarzeń.

Kluczowe elementy diagramu:

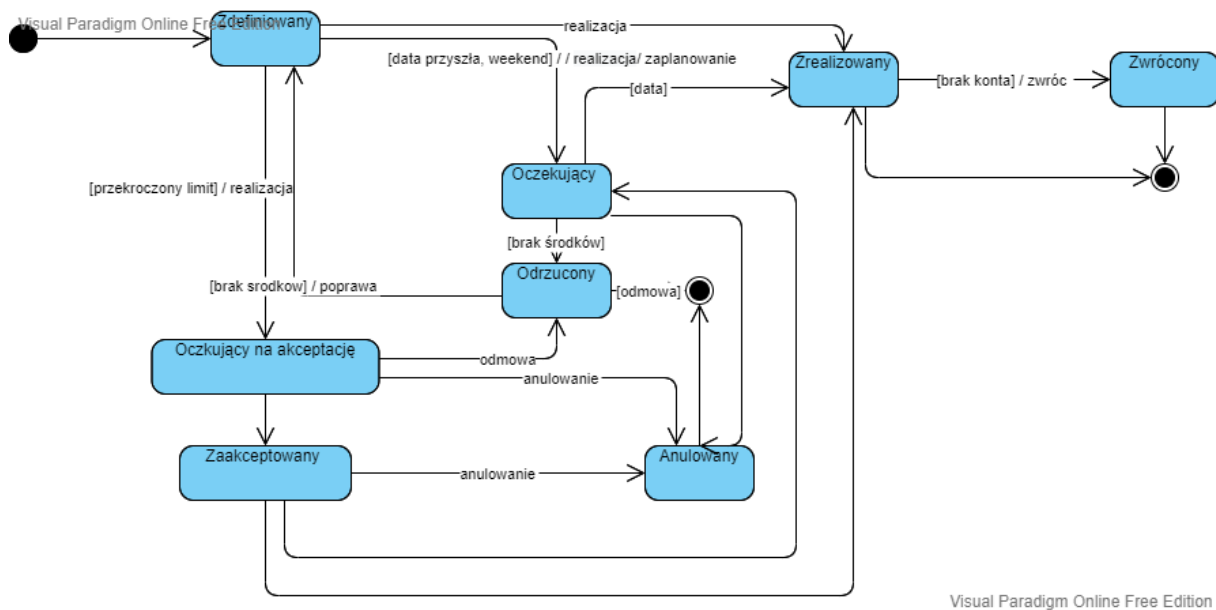
- Stany: Stany reprezentują różne fazy lub warunki, w jakich może znajdować się obiekt lub system. Na diagramie są one reprezentowane jako prostokąty lub elipsy. Każdy stan ma swoją nazwę, która opisuje go w sposób zrozumiały dla modelowanego systemu.
- Przejścia: Przejścia stanowią ścieżki między stanami. Określają one, w jaki sposób obiekt lub system przechodzi z jednego stanu do drugiego w odpowiedzi na zdarzenia. Przejścia są reprezentowane jako strzałki, które łączą stany. Mogą mieć etykiety, które opisują zdarzenie, które wywołuje dane przejście.
- Zdarzenia: Zdarzenia są bodźcami, które wywołują przejście między stanami. Mogą to być sygnały, komunikaty, czasomierze, warunki logiczne itp. Zdarzenia są związane z przejściami i mogą wywołać zmianę stanu obiektu.
- Decyzje: Decyzje (warianty) są używane do modelowania różnych warunków lub scenariuszy przejścia między stanami. Pozwalają na rozgałęzianie i wybór ścieżki na podstawie warunków logicznych lub zdarzeń.

Diagram przejść stanów pozwala zobaczyć jak obiekt lub system zmienia swoje stany w odpowiedzi na zdarzenia. Pokazuje również, jakie akcje są wykonywane podczas przejścia między stanami.



Rysunek 3 Przykład diagramu przejść stanów dla obiektu konto klienta (system bankowy)





Rysunek 4 Przykład diagramu przejść stanów dla obiektu przelew (system bankowy)

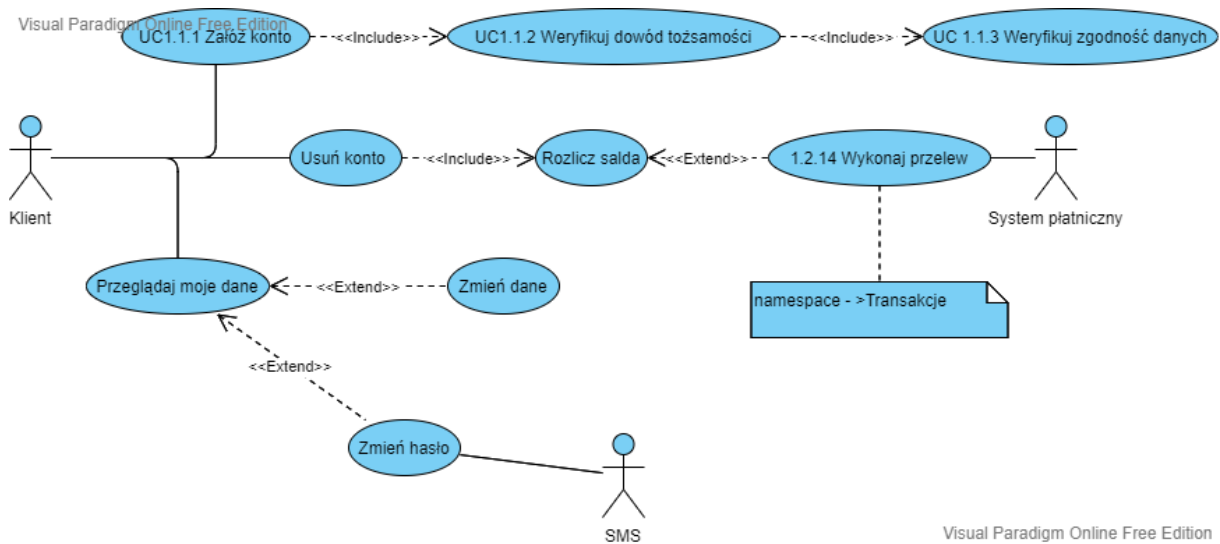
## Diagram przypadków użycia

Diagram przypadków użycia (ang. *use case diagram*) w UML (Unified Modeling Language) jest graficznym narzędziem do modelowania funkcjonalności systemu z perspektywy interakcji między aktorami (użytkownikami zewnętrznymi) a systemem. Jest używany do identyfikacji, opisu i organizacji różnych przypadków użycia w systemie. Diagram przypadków użycia pomaga zrozumieć interakcje między aktorami a systemem oraz identyfikować funkcjonalności, które system powinien obsługiwać.

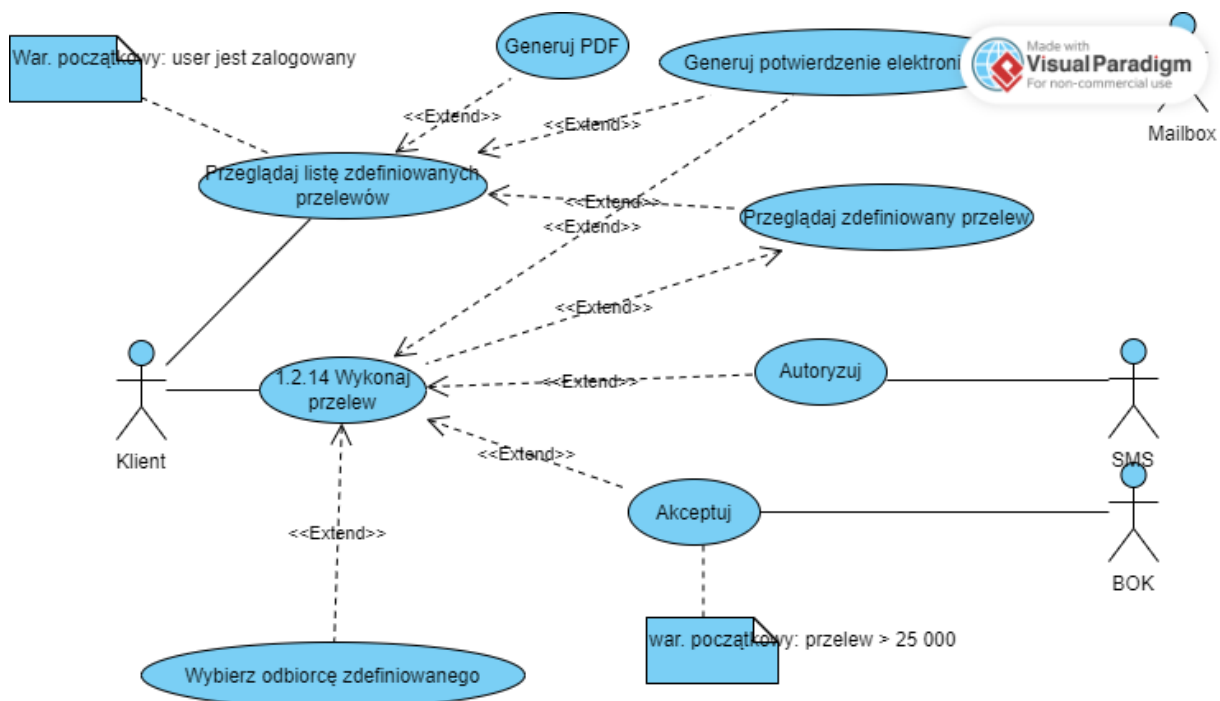
Kluczowe elementy diagramu:

- **Aktorzy:** Aktorzy reprezentują zewnętrzne podmioty, takie jak użytkownicy lub systemy zewnętrzne.
- **Przypadki użycia:** Przypadki użycia reprezentują funkcjonalne wymagania systemu. Są to działania, które mogą zostać wykonane przez aktorów w systemie. Przypadki użycia są reprezentowane jako elipsy na diagramie. Każdy przypadek użycia powinien mieć nazwę, która opisuje jego funkcję.
- **Asocjacje:** Asocjacje są liniami łączącymi aktorów z przypadkami użycia. Wskazują, że dany aktor może korzystać z danego przypadku użycia. Asocjacje mogą mieć również etykiety, które opisują relację między aktorem a przypadkiem użycia.
- **Relacje między przypadkami użycia:** Na diagramie przypadków użycia można również przedstawić relacje między przypadkami użycia. Niektóre z najczęściej używanych relacji to:
  - **Relacja zawierania (include):** Przypadek użycia A zawiera przypadek użycia B, gdy A korzysta z B jako części swojego działania.
  - **Relacja rozszerzania (extend):** Przypadek użycia A rozszerza przypadek użycia B, gdy A rozszerza funkcjonalność B o dodatkowe kroki lub działania.



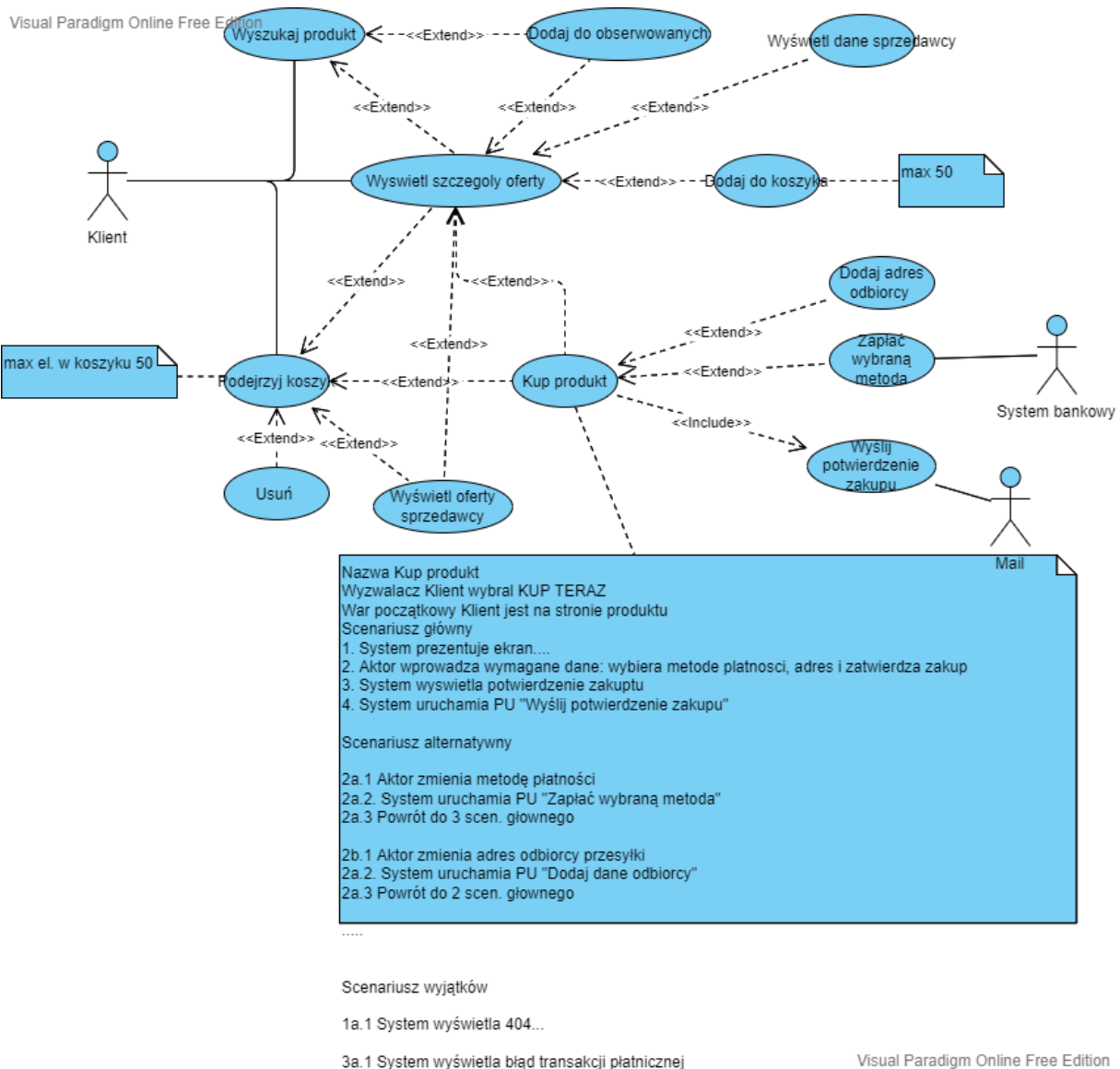


Rysunek 5 Przykład diagramu przypadków użycia dla systemu bankowego



Rysunek 6 Przykład diagramu przypadków użycia dla systemu bankowego - przelewy





Rysunek 7 Przykład diagramu przypadków użycia dla e-sklepu

## Przypadki użycia – zastosowanie

### Dokumentacja funkcjonalności systemu

1. Diagram przypadków użycia
2. Specyfikacja przypadków użycia

### Specyfikacja przypadków użycia

1. Oznaczenie: Unikalny identyfikator przypadku użycia
2. Nazwa: Unikalna nazwa PU określająca jego cel
3. Autorzy: Nazwy autorów zaangażowanych w opis PU
4. Priorytet: Nadana ranga dla PU
5. Krytyczność: Określa jak wiele szkody może wyrządzić niepowodzenia realizacji PU
6. Źródło: Oznaczenie źródła, z którego powodzi PU (np. interesariusz, dokument, system)
7. Osoba odpowiedzialna: Interesariusz, który jest odpowiedzialny za dany PU
8. Opis: Krótki opis PU





9. Wyzwalacze: Nazwa zdarzenia, która powoduje uruchomienie PU
10. Aktorzy: Lista aktorów, którzy wchodzą w interakcję z danym PU
11. Warunki wstępne: Lista warunków, które muszą być spełnione aby PU mógł być uruchomiony
12. Warunki końcowe: Lista stanów systemu jakie muszą być osiągnięte po poprawnym wykonaniu scenariusza głównego PU
13. Rezultat: Opis rezultatów jakie są osiągane podczas realizacji PU
14. Główny scenariusz: Opis scenariusza głównego określającego ścieżkę wykonania PU
15. Scenariusze alternatywne: Opis scenariuszy alternatywnych określających oraz zdarzeń powodujących przejście do realizacji scenariusza alternatywnego
16. Scenariusze wyjątków: Opis scenariuszy dla wyjątków oraz zdarzeń powodujących przejście do realizacji scenariusza wyjątku
17. Wymagania jakościowe: Odsyłacze do wymagań jakościowych, które dotyczą opisywanego PU

### Przykłady specyfikacji przypadku użycia

#### *PYSZ\_UC\_0010 Wyszukaj restaurację*

Id	PYSZ_UC_0010
Nazwa	Wyszukaj restaurację
Aktor	Użytkownik aplikacji mobilnej
Warunki początkowe	- znana jest lokalizacja aktora
Scenariusz główny	<ol style="list-style-type: none"> <li>1. Aktor wpisuje frazę w kryterium wyszukiwania</li> <li>2. W wpisaniu 2 znaków system podpowiada wyniki</li> <li>3. Aktor wybiera pasującą frazę z podpowiedzi</li> <li>4. System wyświetla wyniki spełniające kryterium wyszukiwania</li> </ol>
Scenariusz alternatywny	<p>Do kroku 3. <i>Ręczne wprowadzenie danych</i></p> <ol style="list-style-type: none"> <li>1. Aktor wprowadza pełną nazwę restauracji manualnie</li> <li>2. System wyświetla wyniki spełniające kryterium wyszukiwania</li> </ol> <p>Do kroku 4. <i>Brak wyników</i> System wyświetla komunikat „Nie znaleziono wyników”</p> <p>Do kroku 4. <i>Dużo wyników</i> System wyświetla pierwszych 20 restauracji, kolejnych 20 będzie wyświetlane po przewinięciu strony przez aktora</p>
Scenariusz rozszerzenia	<p>Do kroku 4 <i>Przeglądanie menu</i> Aktor wywołuje PYSZ_UC_0011 <i>Przeglądaj menu</i></p>
Scenariusz wyjątku	<p>Do kroku 4. <i>Błąd połączenia z BD</i> System wyświetla komunikat ‘Nie udało się połączyć z bazą danych, spróbuj ponownie za chwilę”</p>
Warunki końcowe	System wyświetla wyniki zgodne z kryterium wyszukiwania
Gwarancja minimalna	System wyświetla komunikat błędu



PYSZ\_UC\_0015 Złóż zamówienie

Id	PYSZ_UC_0015
Nazwa	Złóż zamówienie
Aktor	Użytkownik aplikacji mobilnej
Warunki początkowe	- istnieje co najmniej 1 pozycja w koszyku - Suma pozycji w koszyku musi być większa lub równa minimalnej kwocie zamówienia określonej przez daną restaurację
Scenariusz główny	<ol style="list-style-type: none"> <li>1. Aktor wybiera opcję składania zamówienia dostępną z pozycji koszyka</li> <li>2. System wyświetla formularz na którym należy podać dane niezbędne do zamówienia (<i>PYSZ_SCR_0015.1 Dane zamówienia</i>)</li> <li>3. Aktor podaje wymagane dane</li> <li>4. System wyświetla sekcję do wyboru metody płatności</li> <li>5. Aktor wybiera metodę płatności i inicjalizuje płatność</li> <li>6. System uruchamia PYSZ_UC_0016 Opłać zamówienie</li> <li>7. System przekazuje zamówienie do restauracji i wyświetla ekran potwierdzenia zamówienia (<i>PYSZ_SCR_0015.2 Potwierdzenie zamówienia</i>)</li> </ol>
Scenariusz alternatywny	<p>Na każdym etapie <i>Anulowanie zamawiania</i></p> <ol style="list-style-type: none"> <li>1. Aktor anuluje proces składania zamówienia wybierając opcję anuluj</li> <li>2. System wyświetla ekran główny</li> </ol> <p>Do kroku 2. <i>Logowanie do systemu</i> Aktor wybiera opcję logowania. System uruchamia PYSZ_UC_0002 Loguj. Po zalogowaniu, dane niezbędne do zamówienia są uzupełnione automatycznie.</p> <p>Do kroku 4 <i>Nieprawidłowe dane zamówienia – błędy format nr telefonu</i> System wyświetla komunikat błędu „Podaj numer telefonu w formacie xxx-xxx-xxx”</p> <p>Do kroku 7 <i>Błąd płatności</i></p> <ol style="list-style-type: none"> <li>1. System wyświetla komunikat „Nie udało się wykonać płatności. Spróbuj ponownie”</li> <li>2. Powrót do kroku 4 scenariusza głównego</li> </ol>
Scenariusz rozszerzenia	
Scenariusz wyjątku	<p>Do kroku 2 i 4. <i>Błąd połączenia z BD</i> System wyświetla komunikat ‘Nie udało się połączyć z bazą danych, spróbuj ponownie za chwilę”</p> <p>Do kroku 7 <i>Błąd systemowy</i> System wyświetla komunikat „Nie ma możliwości złożenia zamówienia. Zwrot płatności nastąpi w ciągu 24 godzin”. System uruchamia <i>PYSZ_UC_0020_Zwróć płatność.</i></p>
Warunki końcowe	System wyświetla komunikat potwierdzenia zamówienia Zamówienie jest przekazane do restauracji
Gwarancja minimalna	System wyświetla komunikat błędu Automatyczny zwrot płatności



## Diagram aktywności

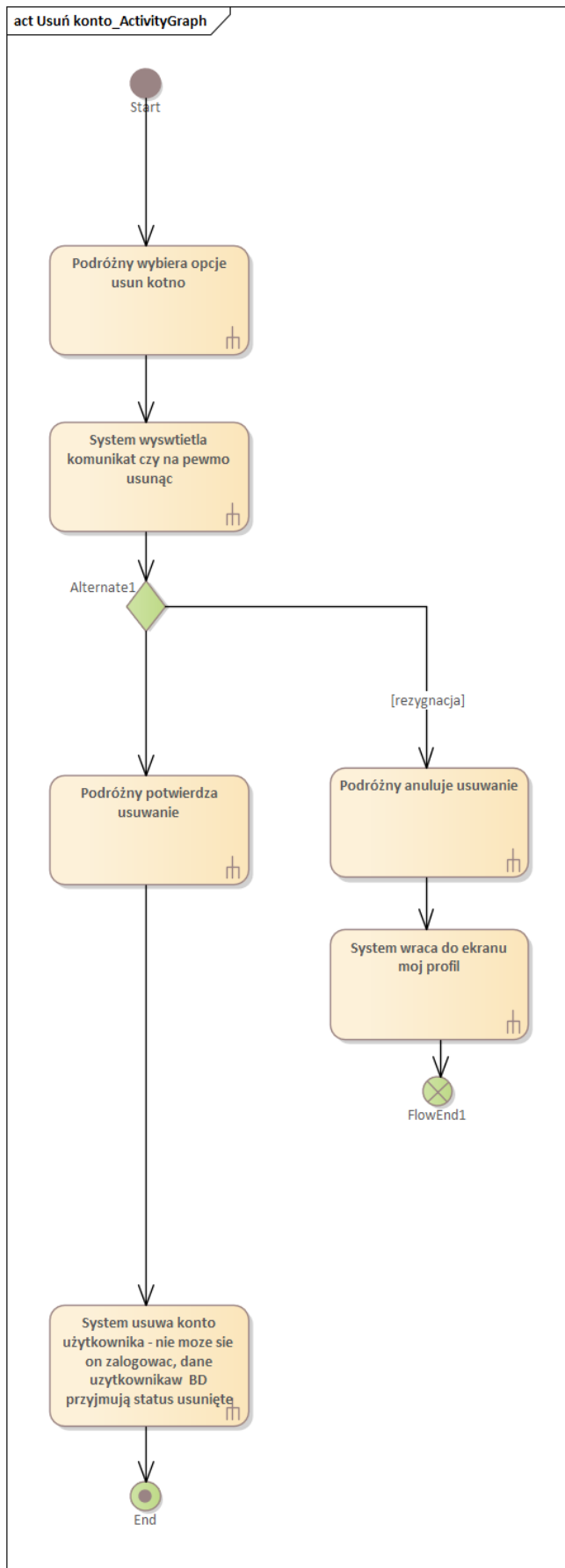
Diagram aktywności (ang. *activity diagram*) w UML (Unified Modeling Language) jest graficznym narzędziem do modelowania dynamicznych aspektów systemu.

Diagram aktywności jest używany do modelowania sekwencji działań, procesów biznesowych, algorytmów lub innych dynamicznych aspektów systemu. Przedstawia interakcje między różnymi czynnikami i pokazuje przepływ sterowania w systemie, co pomaga w zrozumieniu i analizie procesów zachodzących w systemie.

Kluczowe elementy diagramu:

- **Akcje:** Akcje reprezentują konkretne działania, które są podejmowane w trakcie procesu. Mogą to być operacje, obliczenia, decyzje, komunikacje itp. Akcje są reprezentowane jako prostokąty z zaokrąglonymi rogami i zapisywane wewnątrz nich.
- **Strzałki przepływu sterowania:** Strzałki przepływu sterowania pokazują kolejność wykonywania akcji lub przejścia między akcjami. Wskazują kierunek przepływu sterowania między akcjami i określają, która akcja jest wykonywana jako następna.
- **Decyzje:** Decyzje są warunkami logicznymi, które prowadzą do różnych ścieżek wykonania w zależności od spełnienia określonych warunków. Są reprezentowane jako romby i mają strzałki wychodzące, które wskazują na różne akcje w zależności od wyniku decyzji.
- **Rozgałęzienia i łączenia:** Rozgałęzienia (forks) i łączenia (joins) są używane do równoczesnego podziału lub łączenia przepływu sterowania w różnych strukturach decyzyjnych. Pozwalają na równoległe wykonywanie różnych akcji lub równoległe scalanie różnych ścieżek przepływu.
- **Obiekty i role:** Diagram aktywności może również zawierać obiekty lub role, które biorą udział w wykonywanych akcjach. Obiekty te mogą mieć atrybuty lub role, które są istotne dla przepływu aktywności.

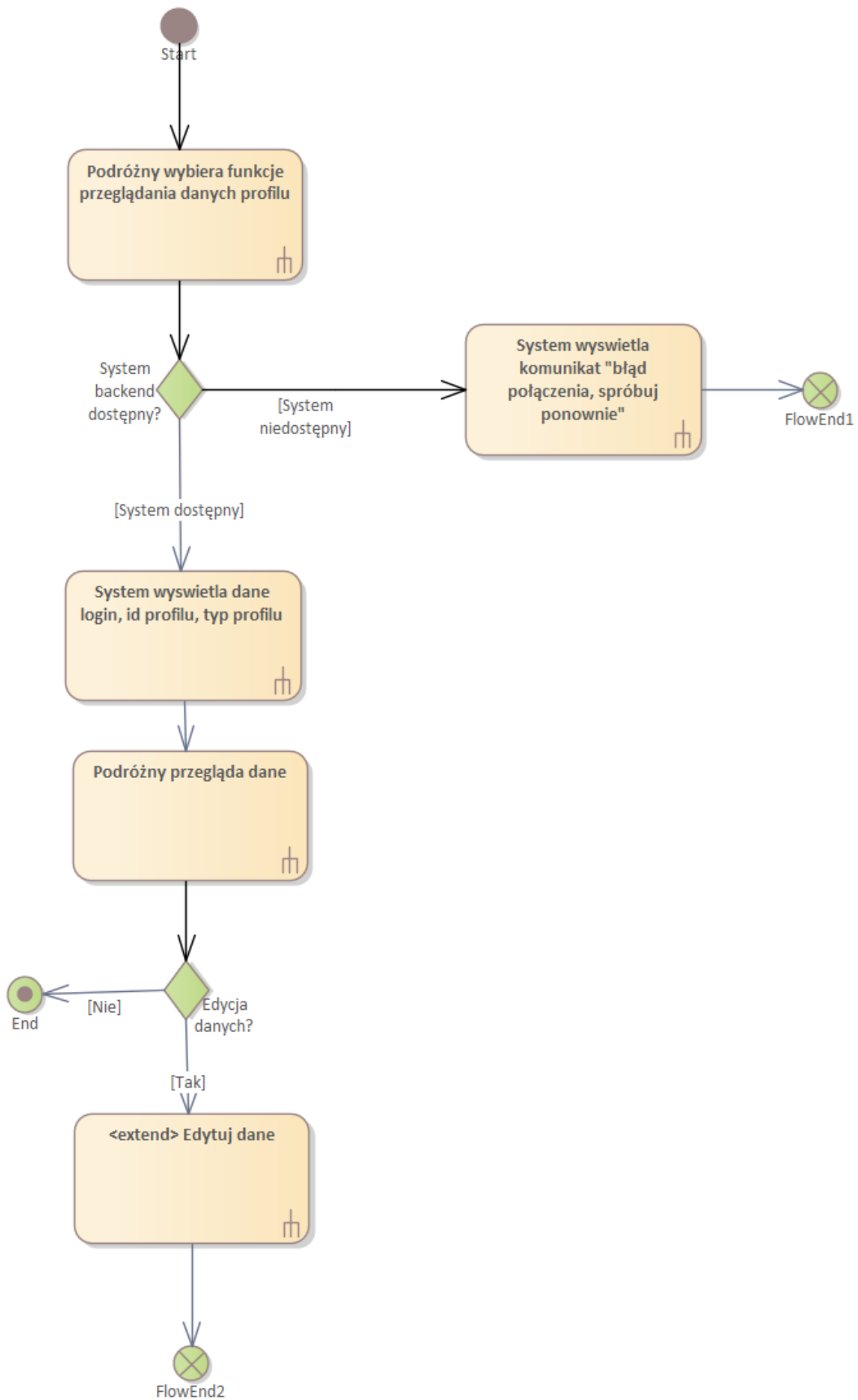




Rysunek 8 Przykład diagramu aktywności - usuwanie konta klienta



act WYświetl dane profilu\_ActivityGraph



Rysunek 9 Przykład diagramu aktywności wyświetlanie danych profilu klienta

Zaloz\_konto

