

Notatki SOB

Prezentacja 1

Klasyfikacja zagrożeń:

1. **Działania człowieka:** Zagrożenia wynikające z działalności człowieka obejmują nieuprawniony dostęp do systemu, kradzież danych, szpiegostwo przemysłowe, uszkodzenie sprzętu, błędy programistyczne, a także celowe działania sabotażowe. Może to dotyczyć zarówno pracowników firmy, jak i osób z zewnątrz, którzy próbują uzyskać dostęp do systemów.
2. **Zdarzenia losowe:** Zagrożenia wynikające z zdarzeń losowych obejmują awarie sprzętu, awarie systemów, błędy programistyczne, awarie zasilania, zdarzenia naturalne, takie jak trzęsienia ziemi, powodzie, huragany, pożary, a także awarie spowodowane przez ludzi, takie jak wypadki samochodowe, pożary budynków, itp.
3. **Zagrożenia wewnętrzne:** Zagrożenia wewnętrzne obejmują nieuprawnione działania pracowników, błędy programistyczne, złośliwe oprogramowanie, niezgodne z polityką firmy działania, a także niedostateczne zabezpieczenia przed zagrożeniami zewnętrznymi.
4. **Zagrożenia zewnętrzne:** Zagrożenia zewnętrzne obejmują ataki ze strony hakerów, wirusy komputerowe, ataki phishingowe, ataki typu DDoS, a także ataki z wykorzystaniem wad w oprogramowaniu. Ataki mogą być prowadzone przez osoby trzecie, grupy przestępcze lub państwa, a ich celem może być kradzież informacji, uszkodzenie systemów, szantażowanie lub szpiegostwo przemysłowe.

Typy zagrożeń:

Zagrożenia w sieci komputerowej:

1. Wirusy komputerowe - oprogramowanie, które może się replikować i rozprzestrzeniać się w sieci, powodując uszkodzenia systemu, kradzież danych lub zmuszenie do zapłaty okupu za odzyskanie danych.
2. Ataki DDoS - próby sparaliżowania działania systemu przez przeciążenie serwera nadmierną ilością zapytań z różnych źródeł jednocześnie.
3. Ataki typu phishing - próby oszukania użytkowników, aby podali swoje poufne dane, takie jak hasła lub informacje karty kredytowej, zazwyczaj za pośrednictwem fałszywych stron internetowych lub e-maili.
4. Ataki typu man-in-the-middle - próby przechwycenia i zmiany przesyłanych danych między dwoma komputerami, w celu uzyskania poufnych informacji lub wykonania fałszywych transakcji.
5. Nieuprawniony dostęp do systemu - próby uzyskania dostępu do systemu przez osoby trzecie, które nie mają uprawnień do korzystania z niego, w celu kradzieży danych lub przeprowadzenia innych działań sabotażowych.

Zagrożenia w systemach komputerowych:

1. Błędy programistyczne - niedoskonałości w kodzie źródłowym, które mogą prowadzić do niestabilności systemu lub poważnych uszkodzeń.
2. Uszkodzenie sprzętu - awarie lub błędy w sprzęcie, takie jak dyski twarde, pamięci RAM, zasilacze, itp., które mogą prowadzić do utraty danych lub niestabilności systemu.
3. Ataki typu exploit - próby wykorzystania podatności w systemie lub oprogramowaniu, aby zdobyć nieuprawniony dostęp do systemu lub uzyskać kontrolę nad nim.
4. Złośliwe oprogramowanie - oprogramowanie, które zostało stworzone w celu szkodenia systemowi lub kradzieży danych, takie jak wirusy, trojany, ransomware, itp.
5. Nieuprawnione działania użytkowników - działania pracowników lub innych użytkowników systemu, które są niezgodne z polityką firmy lub normami etycznymi, takie jak kradzież danych, uszkodzenie sprzętu, itp.

Cele bezpieczeństwa komputerowego

- poufność
- nienaruszalność
- dostępność

NIST FIPS 199 (Federal Information Processing Standards Publication 199) to amerykański standard dotyczący klasyfikacji informacji oraz określania poziomów bezpieczeństwa systemów informatycznych. Standard ten został opracowany w celu zdefiniowania wymagań bezpieczeństwa dla systemów informatycznych używanych przez rząd USA.

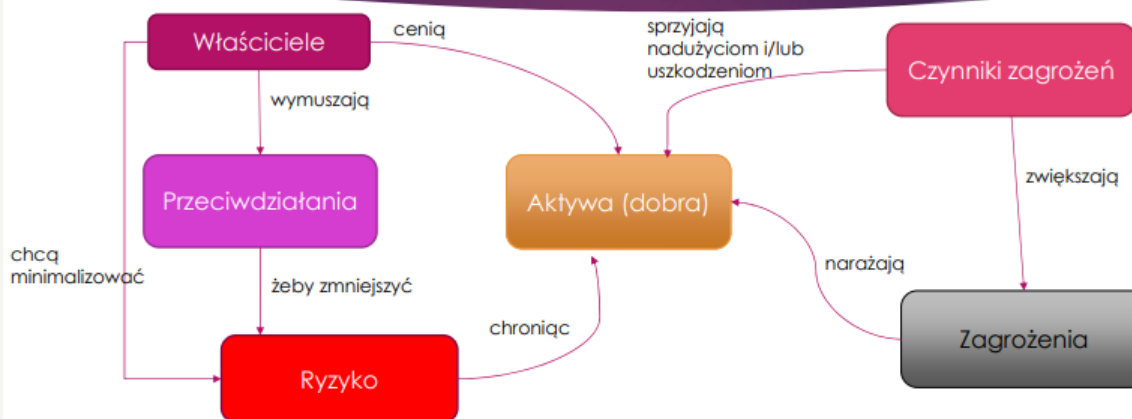
FIPS 199 definiuje trzy poziomy klasyfikacji informacji: niski, średni i wysoki. Klasyfikacja ta opiera się na trzech kryteriach:

- Poufność - określa, jakie informacje powinny być chronione przed nieuprawnionym dostępem i jakie są skutki w przypadku ich ujawnienia.
- Integralność - określa, jakie informacje powinny być chronione przed modyfikacją lub usunięciem i jakie są skutki w przypadku ich zmiany.
- Dostępność - określa, jakie informacje powinny być dostępne dla uprawnionych użytkowników i jakie są skutki w przypadku ich braku.

W zależności od tych kryteriów, FIPS 199 definiuje minimalne wymagania bezpieczeństwa dla każdego poziomu klasyfikacji. Niski poziom klasyfikacji wymaga minimalnego poziomu kontroli bezpieczeństwa, a wysoki poziom wymaga najwyższego poziomu kontroli.

Standard NIST FIPS 199 jest często stosowany w sektorze publicznym, a także w sektorze prywatnym jako wytyczna dla określenia wymagań bezpieczeństwa dla systemów informatycznych.

Model bezpieczeństwa komputerowego



	Dostępność	Poufność	Nienaruszalność
sprzęt	Wypożyczenie zostało skradzione lub unieruchomione uniemożliwiając świadczenie usług	Zdeszyfrowany napęd (jednostka pamięci) USB	
oprogramowanie	Usunięto programy, co uniemożliwia dostęp do nich użytkownikom	Wykonano oprogramowania upoważnienia kopię bez	Działający program jest modyfikowany, aby spowodować awarię podczas wykonania lub wykonać niepożądane zadanie
dane	Następuje usunięcie plików, co uniemożliwia dostęp do nich użytkownikom	Wykonano nielegalne czytanie danych. Analiza statystyczna danych ujawnia głębsze dane.	Istniejące pliki zostały zmodyfikowane lub sfabrykowane nowe pliki.
Linie i sieci komunikacyjne	Komunikaty zostają zniszczone lub usunięte. Linie komunikacyjne lub sieci stają się niedostępne.	Następuje czytanie komunikatów. Obserwowana charakterystyka komunikatów. jest ruchu	Komunikaty są modyfikowane, usuwane, zmieniona zostaje ich kolejność lub ulegają podwojeniu. Dochodzi do fabrykowania fałszywych komunikatów.

Strategia bezp. Komp.

- Specyfikacja oraz polityka
- Implementacja oraz mechanizmy
- Poprawność oraz pewność
- **Cele:**
 - Ochrona zasobów
 - Uwierzytelnianie
 - Nadawanie uprawnień
 - Integralność
 - danych
 - systemu
 - Nieodrzućenie
 - Poufność
 - Kontrolowanie

Cykl niezawodności i odporności na błędy:

1. Odporność na błędy
2. Efektywne usuwanie
3. minimalizacja w specyfikacji
4. specjalne metody projektowania
5. weryfikacja i testowanie

Metryki złożoności oprogramowania to narzędzia służące do określania poziomu skomplikowania danego systemu lub aplikacji informatycznej. Ich celem jest pomiar ilościowej wartości złożoności oprogramowania w celu oceny jakości, wydajności oraz zarządzania projektem.

- **Cyclomatic Complexity** - to miara złożoności strukturalnej kodu. Określa ona liczbę dróg decyzyjnych w kodzie, czyli takich miejsc, w których program musi podjąć decyzję o wyborze jednego z dwóch lub więcej możliwych kierunków. Im wyższa wartość Cyclomatic Complexity, tym większe prawdopodobieństwo wystąpienia błędów w kodzie.
- **Lines of Code** - to najprostsza metryka, która określa liczbę linii kodu w projekcie. Jednakże sama ilość linii kodu nie zawsze odzwierciedla poziom złożoności kodu i może być myląca. Na przykład, kod można napisać bardziej zwięźle, używając mniej linii kodu, ale jednocześnie wykorzystując bardziej skomplikowane konstrukcje językowe.

- **Function Points** - to miara złożoności funkcjonalnej systemu, która określa ilość funkcji, które dany system wykonuje. Punkty funkcjonalne uwzględniają między innymi złożoność interfejsów użytkownika, logikę biznesową, a także charakter danych, z którymi system pracuje. Funkcje są oceniane pod kątem ich trudności i wpływu na inne elementy systemu. Punkty funkcjonalne pozwalają na ocenę złożoności systemu z perspektywy użytkownika i stanowią pomocne narzędzie do zarządzania projektem oraz określania kosztów wytwarzania oprogramowania.

Metryki złożoności oprogramowania są ważne dla oceny jakości kodu i poprawy procesów programistycznych. Ich wykorzystanie może pomóc w identyfikacji problemów związanych z złożonością kodu oraz ułatwić ich rozwiązanie.

W kontekście systemów informatycznych, terminy "**defekt**", "**awaria**" i "**błąd**" mają zwykle następujące znaczenie:

- Defekt - to ogólne pojęcie oznaczające niedoskonałość lub nieprawidłowość w oprogramowaniu lub sprzęcie komputerowym. Defekt może obejmować błędy, niedociągnięcia, luki w zabezpieczeniach lub inne problemy.
- Awaria - to stan, w którym system informatyczny nie działa zgodnie z oczekiwaniami lub przestaje działać w ogóle. Awaria może mieć różne przyczyny, w tym błędy programistyczne, problemy z siecią, błędy w sprzęcie lub problemy z zasobami systemowymi.
- Błąd - to szczególny rodzaj defektu, który powoduje nieprawidłowe działanie systemu lub jego części. Błędy mogą wynikać z różnych przyczyn, takich jak błędne algorytmy, niewłaściwe dane wejściowe lub problemy z pamięcią.

W skrócie, defekt jest ogólnym pojęciem odnoszącym się do niedoskonałości w systemie informatycznym, awaria to stan, w którym system przestaje działać zgodnie z oczekiwaniami, a błąd to konkretny rodzaj defektu, który powoduje nieprawidłowe działanie systemu lub jego części.

Metryki w apkach webowych

- ważona liczba metod na klasie
- głębokość drzewa dziedziczenia
- liczba potomków w drzewie dziedziczenia
- związki między klasami
- brak spójności między metodami

Zapobieganie defektom

- UNIKANIE DEFEKTÓW
 - rygorystyczne i/lub formalne specyfikacje wymagań
 - zastosowanie sprawdzonych metod projektowania
 - użycie języków z mechanizmami wspierającymi abstrakcje, weryfikacje
 - użycie narzędzi IP
- USUWANIE DEFEKTÓW
 - weryfikacja
 - walidacja
 - testowanie

Testowanie oprogramowania

- Wybór losowy
- Pokrycie wymagań
- Testowanie white-box (rodzaj testów w inżynierii oprogramowania, polegających na testowaniu programu poprzez podawanie na wejściu takich danych, aby program przeszedł przez każdą zaimplementowaną ścieżkę.)
- Wybór oparty na modelu
- Mechanizm tolerancji defektów
- Systemy cykliczne

Formułowanie hipotezy defektów to proces tworzenia przewidywań na temat potencjalnych błędów i problemów w systemie, na podstawie analizy danych z różnych źródeł. W tym procesie ważne jest uwzględnienie doświadczenia i wiedzy specjalistów z danej dziedziny, a także stosowanie odpowiednich narzędzi i metodyk. Hipotezy defektów mogą pomóc w zapobieganiu błędom, poprawie jakości systemu oraz zoptymalizowaniu procesów programistycznych.

Systemy rozproszone to złożone aplikacje, w których różne elementy są rozproszone na wiele niezależnych komputerów połączonych w sieć. Pozwala to na równoległe przetwarzanie dużej ilości danych oraz zapewnia skalowalność i niezawodność systemu. Jednym z wyzwań związanym z systemami rozproszonymi jest zapewnienie spójności i synchronizacji danych pomiędzy różnymi komponentami systemu.

Głosowanie rozproszone: proponowana zmiana zostaje przyjęta jeśli większość węzłów się zgodzi

Prezentacja 2

Czym jest system biznesowy

- połączenie elementów dynamicznych (proces biznesowy) i statycznych (obiekty biznesowe, informacyjne, struktura organizacyjna)
- procedura lub proces który jest stosowany jako mechanizm dostarczania konkretnych towarów lub usług
- procedura, proces, metoda zaprojektowany w celu osiągnięcia wyniku. Jego części i etapy współgrają. ...

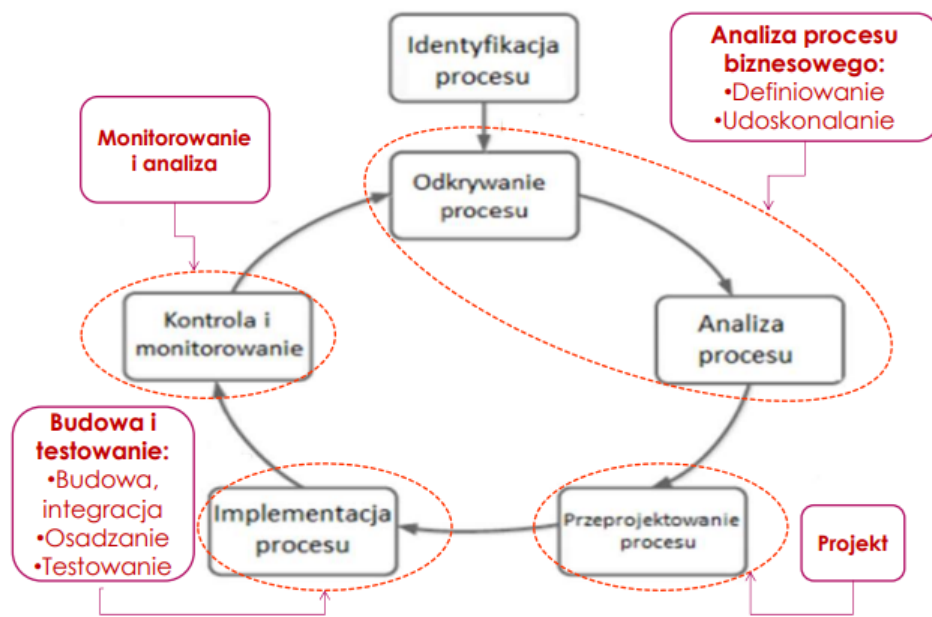
ZACHOWANIE SPÓJNOŚCI

- **model systemu biznesowego**
 - oferowane korzyści
 - strategię dostarczania
 - obszar biznesu
 - koszt, czas
- **model systemu informatycznego**
 - funkcjonalność
 - struktura systemu
 - architektura
 - dynamika systemu

Proces biznesowy to sekwencja powiązanych ze sobą działań podejmowanych w celu osiągnięcia określonego celu biznesowego. Mierniki procesu biznesowego to narzędzia służące do oceny efektywności i efektywności procesu, takie jak:

- czas trwania,
- koszty,
- jakość,
- zadowolenie klienta
- elastyczność
- terminowość
- znaczenie dla organizacji i klienta
- wydajność.

Cykl życia procesu biznesowego



BPMN(Business Process Model and Notation)

- Proces przedstawiany w formie diagramu
- **Diagramy :**
 - procesu
 - współpracy
 - konwersacji
 - choreografii

Główne cele to :

- wizualizacja projektu
- dokumentacja
- komunikacja

???

Proces wewnętrzny dotyczy wewnętrznych działań organizacji, takich jak produkcja, dostawa czy zarządzanie zasobami ludzkimi.

Proces zewnętrzny obejmuje interakcje organizacji z klientami, dostawcami lub partnerami biznesowymi.

Proces kooperacji (B2B) odnosi się do procesów, w których dwie lub więcej organizacji współpracuje w celu realizacji wspólnego celu.




ELEMENTY BPMN

1. **Zdarzenia** (Events) - reprezentują momenty w czasie, które zmieniają stan procesu biznesowego, takie jak start, koniec, błędy, alarmy lub inne zdarzenia, które wymagają reakcji.
2. **Działania** (Activities) - reprezentują zadania, które muszą być wykonane w ramach procesu biznesowego, takie jak przetwarzanie danych, decyzje biznesowe, przepływ informacji lub inne czynności wymagające działania.
3. **Bramki** (Gateways) - reprezentują punkty decyzyjne w procesie biznesowym, które określają, którą ścieżką procesu powinna być wybrana w zależności od warunków biznesowych.
4. **Sekwencje** (Sequence Flows) - reprezentują połączenia pomiędzy elementami procesu biznesowego, określają kolejność wykonywania działań.
5. **Powiązania komunikaty** (Message Flows) - reprezentują przepływ informacji lub wiadomości pomiędzy elementami procesu biznesowego, określają interakcje między różnymi podmiotami w procesie biznesowym.
6. **Baseny** (Pools) - reprezentują podmioty biznesowe lub grupy, które wykonują aktywności w ramach procesu biznesowego.
7. **Tory pływackie** (Lanes) - reprezentują grupy zadań lub ról, które wykonują aktywności w ramach procesu biznesowego.
8. **Dane** (Data) - reprezentują informacje przetwarzane i używane w ramach procesu biznesowego.
9. **Adnotacje** (Annotations) - służą do opisu elementów procesu biznesowego, takich jak opisy, komentarze lub informacje dodatkowe.
10. **Artefakty** (Artifacts) - są to elementy pomocnicze służące do ułatwienia czytelności i zrozumienia procesu biznesowego, takie jak ikony, symbole lub wykresy.

Zadanie

- prostokąt z zaokrąglonymi rogami
- praca w procesie
- jeśli jest znacznik [+] na symbolu, oznacza to że jest to podproces

Wybrane typy czynności w BPMN

 Przyjęcie zapłaty w gotówce	Czynność ręczna (ang. manual) – czynność wykonywana w całości przez człowieka, bez wykorzystania systemu informatycznego
 Zarejestrowanie wpłaty	Czynność użytkownika (ang. user) – czynność wykonywana przez człowieka z użyciem systemu informatycznego
 Obliczenie wysokości odsetek	Czynność automatyczna (ang. service) – czynność wykonywana całkowicie automatycznie, bez udziału człowieka

Zdarzenia i typy zdarzeń

- ❑ **Zdarzenie** (ang. event) - symbolizowane przez okrąg.
- ❑ Punkt początku / końca procesu, pokazanie zmiany stanu w procesie.

zdarzenia inicjujące	zdarzenia pośrednie	zdarzenia końcowe
		

Żeton – token

- ▶ abstrakcyjny element do opisania zachowania procesu
- ▶ „krąży” zgodnie z przepływem sekwencji w procesie
- ▶ „przechodzi” przez elementy modelu BPMN
- ▶ Początek procesu biznesowego generuje żeton, koniec procesu oznacza jego usunięcie
- ▶ Żetony mogą być kreowane, usuwane, rozszczepiane oraz scalane

<http://amber.zarz.agh.edu.pl/amaciol/IT12.pdf>

Wzorce projektowe (design patterns) to sprawdzone rozwiązania problemów projektowych, które zostały udokumentowane i udostępnione dla programistów jako wzorce procesowe. Używanie wzorców projektowych przyspiesza proces projektowania, poprawia jego jakość i ułatwia utrzymanie kodu.

Wzorce procesowe są sprawdzonymi rozwiązaniami, które pomagają projektować, implementować i optymalizować procesy biznesowe.

- **Podstawowe wzorce** to podstawowe elementy, które mogą być wykorzystane w wielu procesach, takie jak sekwencje lub pętle.
- **Zaawansowane wzorce** obejmują bardziej złożone koncepcje, takie jak równoległe przetwarzanie lub dystrybucję zadań.
- **Wzorce strukturalne** dotyczą organizacji procesów, na przykład procesu podziału na mniejsze podprocesy.
- **Wzorce anulowania i stanów** obejmują modele zarządzania sytuacjami wyjątkowymi w procesie.
- **Wzorce obejmujące wiele instancji procesu** są stosowane w sytuacjach, w których wiele instancji procesu jest wykonywanych jednocześnie lub w różnym czasie.

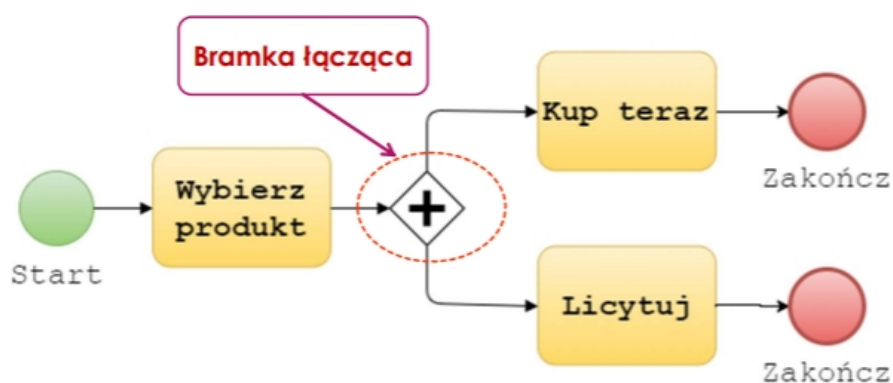
Prezentacja 3

Anomalie w BPMN to sytuacje, w których proces biznesowy nie działa zgodnie z założeniami lub pojawiają się w nim problemy. Przykłady anomalii to: zapętlenie procesu, brak określonej kolejności działań, niespójności w przepływie danych lub brak reakcji na błędy. Anomalie te mogą prowadzić do poważnych problemów związanych z wykonaniem procesu i dlatego należy dążyć do ich minimalizacji poprzez odpowiednie projektowanie, testowanie i wdrażanie procesów biznesowych.

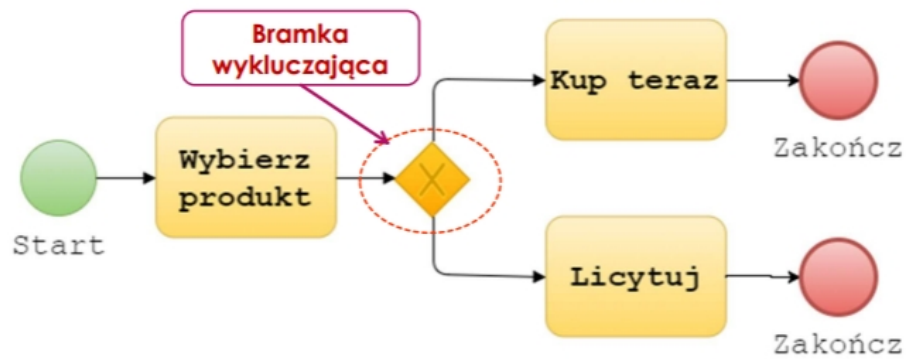
Znane rodzaje:

- **Anomalie składniowe** - to błędy w tworzeniu składni BPMN, takie jak nieprawidłowo użycie znaków lub błędne nazwy elementów.
- **Anomalie strukturalnej** - to błędy w budowie struktury procesu, takie jak brak powiązań między elementami lub nieprawidłowe ułożenie elementów.
- **Wada**

Anomalia: zakleszczenie przepływu procesu

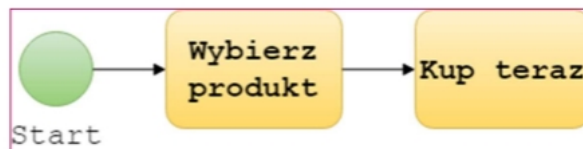


Definiowanie przepływu procesu za pomocą bramki



Anomalia: nieprawidłowe zdarzenie procesowe

Regularny proces BPMN: bez wyraźnego początku i końca.



Poprawne rozwiązanie: użycie **Start** i **Stop**; każdy proces zaczyna się i musi się skończyć.

