

Dokumentacja

System Zarządzania Otwartymi Danymi Publicznymi

Specyfikacja wymagań

Informacje podstawowe

Autorzy

- Grochowski Albert
- Biwejnis Oskar
- Minich Marharyta
- Rohoutseu Ilya
- Rusinovich Ivan
- Yarmats Stanislau

Abstrakt

Dokument zawiera specyfikację wymagań dla systemu Zarządzania Otwartymi Danymi Publicznymi składającą się z wizji systemu, wymagań użytkownika oraz wymagań oprogramowania dla pierwszych iteracji projektu.

Historia zmian

Wersja	Data	Kto	Opis
1.0.0	01.03.2025	Albert Grochowski	Wersja startowa
1.1.0	07.03.2025	Oskar Biwejnis, Minich Marharyta	Dodano procesy biznesowe
1.2.0	14.03.2025	Rohoutseu Ilya, Rusinovich Ivan, Yarmats Stanislau	Dodano wizje systemu

1. Wprowadzenie

1.1 Cel dokumentu

Tutaj znajdzie się opis celu dokumentu... - uzupełnić pod koniec projektu

1.2 Streszczenie dla kierownictwa

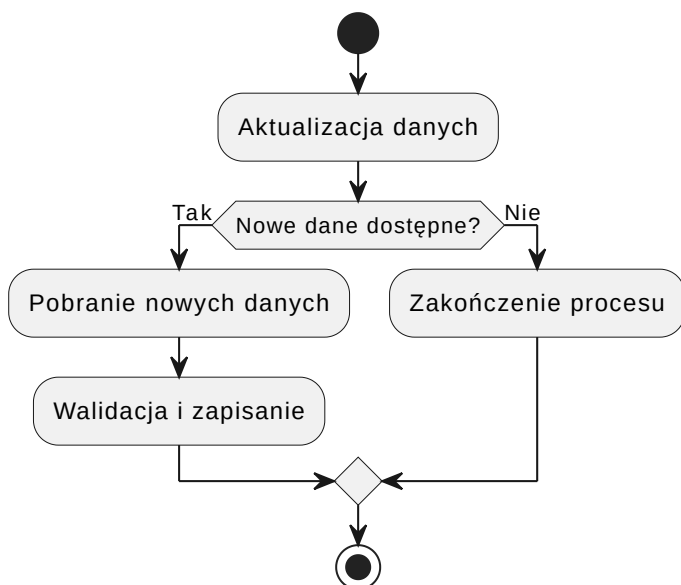
Tutaj znajdzie się streszczenie dla kierownictwa... - uzupełnić pod koniec projektu

2. Opis biznesu

2.1 Procesy biznesowe

(PB0001) Regularna aktualizacja danych

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **regularną aktualizacją danych**. Proces ten składa się z czterech głównych etapów, w których uczestniczy jedna rola:

1. Sprawdzenie dostępności nowych danych

- **Rola:** System
- **Opis:** System weryfikuje, czy nowe dane są dostępne do pobrania z odpowiedniego źródła.

2. Pobranie nowych danych

- **Rola:** System
- **Opis:** Jeśli nowe dane są dostępne, system pobiera je z odpowiedniego źródła. Może to obejmować połączenie z zewnętrznymi bazami danych lub serwisami, aby uzyskać aktualne informacje.

3. Walidacja i zapisanie danych

- **Rola:** System
- **Opis:** System sprawdza poprawność nowych danych, aby upewnić się, że są one zgodne z wymaganiami jakościowymi. Po pomyślnym zwalidowaniu, dane są zapisywane w bazie danych, co zapewnia ich aktualność.

4. Zakończenie procesu

- **Rola:** System
- **Opis:** Jeśli nie ma nowych danych do pobrania, proces kończy się. System nie przeprowadza żadnych dalszych działań.

Proces biznesowy polega na regularnym monitorowaniu dostępności nowych danych oraz ich pobieraniu i aktualizacji w systemie. Celem tego procesu jest utrzymanie bazy danych w stanie aktualności i zgodności z rzeczywistością.

Kod diagramu

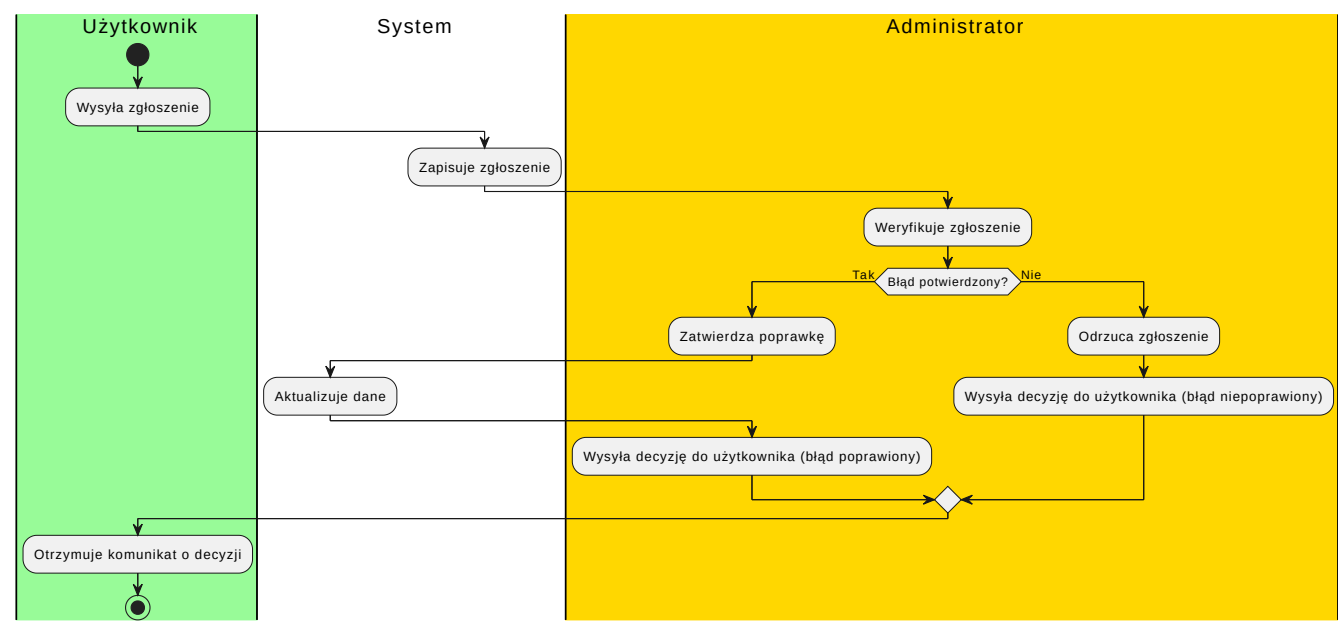
```
@startuml
start

:Aktualizacja danych;
if (Nowe dane dostępne?) then (Tak)
  :Pobranie nowych danych;
  :Walidacja i zapisanie;
else (Nie)
  :Zakończenie procesu;
endif

stop
@enduml
```

(PB0002) Zgłaszanie błędów danych przez użytkowników

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **zgłaszaniem błędów danych przez użytkowników**. Proces ten składa się z kilku głównych etapów, w których uczestniczą trzy role:

1. Zgłoszenie błędu

- Rola: Użytkownik
- Opis: Użytkownik wysyła zgłoszenie o błędzie w danych, które mogą dotyczyć niepoprawnych, niekompletnych lub błędnych informacji znajdujących się w systemie.
- Kolor: #palegreen
- Oznaczenie: u

2. Zapisanie zgłoszenia

- Rola: System
- Opis: System rejestruje zgłoszenie w bazie danych, co pozwala na dalsze śledzenie zgłoszonego problemu i jego rozwiązanie.
- Oznaczenie: s

3. Weryfikacja przez administratora

- Rola: Administrator
- Opis: Administrator sprawdza zgłoszenie, analizując jego zasadność i określając, czy rzeczywiście występuje błąd w danych. Administrator podejmuje decyzję na podstawie dostępnych informacji.
- Kolor: #gold
- Oznaczenie: a

4. Decyzja

- Rola: Administrator
- Opis: Administrator podejmuje decyzję:
 - Jeśli błąd jest potwierdzony, administrator zatwierdza poprawkę.
 - Jeśli błąd nie jest potwierdzony, administrator odrzuca zgłoszenie.
- Kolor: #gold
- Oznaczenie: a

5. Aktualizacja danych (jeśli błąd został potwierdzony)

- Rola: System
- Opis: System aktualizuje dane, aby były poprawne.
- Oznaczenie: s

6. Informacja zwrotna dla użytkownika

- Rola: Użytkownik
- Opis: Użytkownik otrzymuje komunikat o decyzji administratora dotyczącej zgłoszonego błędu.
- Kolor: #palegreen
- Oznaczenie: u

Proces biznesowy polega na umożliwieniu użytkownikom zgłaszania błędów w danych, co pozwala na utrzymanie jakości informacji w systemie. Administratorzy mają za zadanie zweryfikować zgłoszenia i podjąć odpowiednie działania, aby zapewnić poprawność danych.

Kod diagramu

```
@startuml
|#palegreen|u| Użytkownik
start
:Wysyła zgłoszenie;

|s| System
:Zapisuje zgłoszenie;

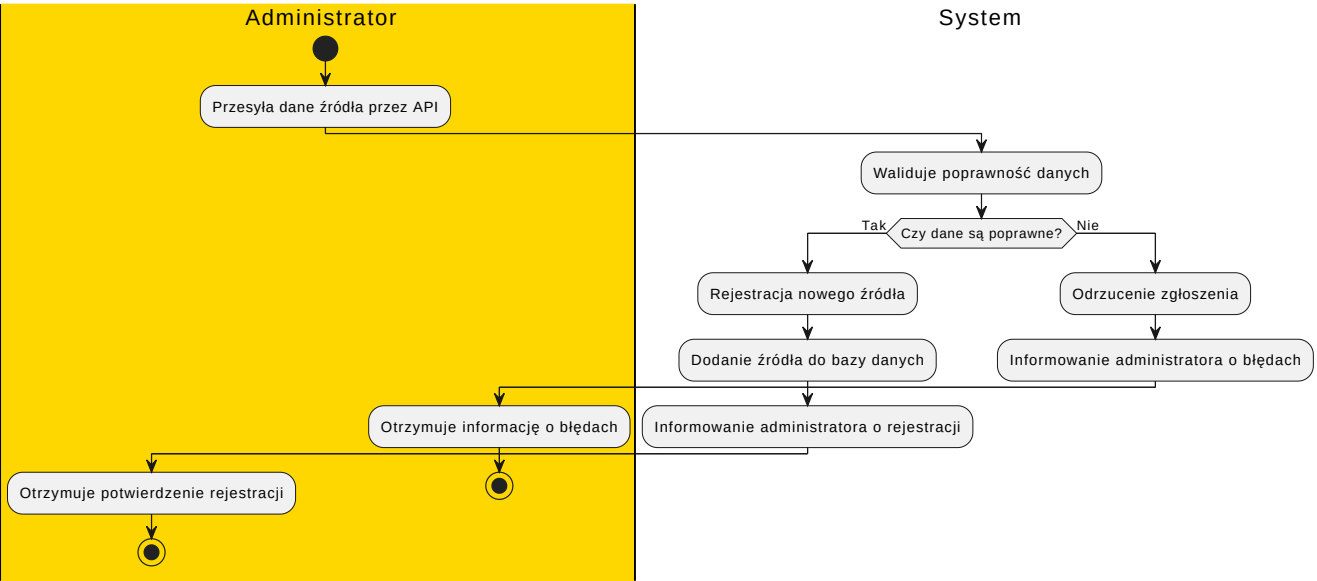
|#gold|a| Administrator
:Weryfikuje zgłoszenie;

if (Błąd potwierdzony?) then (Tak)
  :Zatwierdza poprawkę;
|s| System
:Aktualizuje dane;
|#gold|a| Administrator
:Wysyła decyzję do użytkownika (błąd poprawiony);
else (Nie)
  :Odrzuca zgłoszenie;
  :Wysyła decyzję do użytkownika (błąd niepoprawiony);
endif

|#palegreen|u| Użytkownik
:Otrzymuje komunikat o decyzji;
stop
@enduml
```

(PB0003) Rejestrowanie nowych źródeł danych poprzez api

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **rejestrowaniem nowych źródeł danych poprzez API**. Proces ten składa się z trzech głównych etapów, w których uczestniczą trzy różne role:

1. Przesyłanie danych przez administratora (Send Data by Administrator):

- Rola: Administrator (administrator)
- Opis: Administrator przesyła dane nowego źródła do systemu poprzez API. Dane te mogą obejmować różne informacje o źródle, takie jak adres URL, parametry dostępu i inne niezbędne informacje.
- Kolor: #gold
- Oznaczenie: a

2. Walidacja danych (Data Validation):

- Rola: System (system)
- Opis: System sprawdza poprawność przesyłanych danych, weryfikując, czy spełniają one określone wymagania, takie jak format, kompletność i zgodność z innymi danymi w systemie.
- Oznaczenie: s

3. Rejestracja nowego źródła (Register New Source):

- Rola: System (system)
- Opis: Jeśli dane są poprawne, nowe źródło zostaje zapisane w bazie danych. Jeśli walidacja nie powiedzie się, zgłoszenie zostaje odrzucone, a administrator zostaje poinformowany o błędach.
- Oznaczenie: s

Proces biznesowy umożliwia administratorowi rejestrowanie nowych źródeł danych w systemie w sposób zautomatyzowany i bezpieczny. Dzięki API proces rejestracji jest szybki i efektywny, a walidacja danych zapewnia, że tylko poprawne informacje trafiają do systemu, co poprawia jakość danych i bezpieczeństwo operacji.

Kod diagramu

```
@startuml
|#gold|a| Administrator
start
:Przesyła dane źródła przez API;

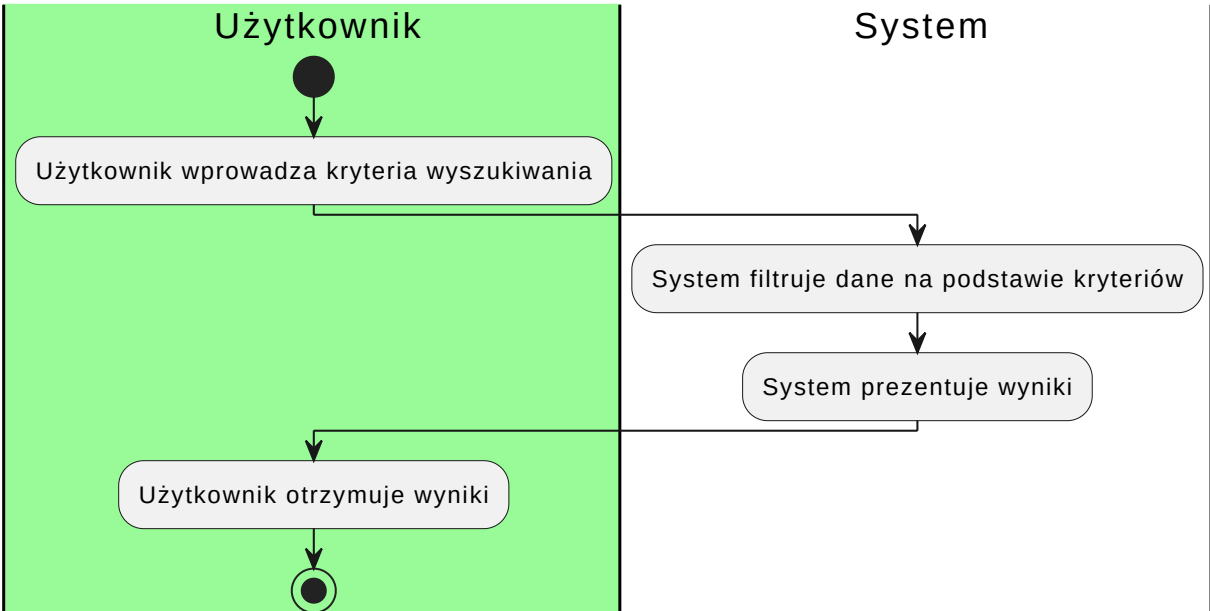
|s| System
:Waliduje poprawność danych;

if (Czy dane są poprawne?) then (Tak)
:Rejestracja nowego źródła;
:Dodanie źródła do bazy danych;
:Informowanie administratora o rejestracji;
|#gold|a| Administrator
:Otrzymuje potwierdzenie rejestracji;
stop
```

```
else (Nie)
  |s| System
  :Odrzucenie zgłoszenia;
  :Informowanie administratora o błędach;
  |#gold|a| Administrator
  :Otrzymuje informację o błędach;
stop
endif
@enduml
```

(PB0004) Filtrowanie wyszukiwanie danych

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **filtrowaniem wyszukiwania danych**:

1. Podanie kryteriów wyszukiwania

- Rola: Użytkownik (user)
- Opis: Użytkownik wprowadza filtry i kryteria wyszukiwania, które pozwolą na określenie, jakie dane mają zostać znalezione.
- Kolor: #palegreen
- Oznaczenie: u

2. Filtrowanie danych

- Rola: System (system)

- Opis: System przetwarza zapytanie użytkownika, przeszukując dostępne dane i stosując odpowiednie filtry, by znaleźć pasujące wyniki.
- Oznaczenie: S

3. Prezentacja wyników

- Rola: System (system)
- Opis: Po przetworzeniu zapytania system wyświetla użytkownikowi wyniki wyszukiwania w odpowiednim formacie, który może być dostosowany do potrzeb użytkownika (np. tabela, lista, grafika).
- Oznaczenie: S

Proces biznesowy polega na zapewnieniu użytkownikowi łatwego dostępu do precyzyjnych informacji w oparciu o zadane kryteria. Wspomaga to podejmowanie decyzji, umożliwiając szybkie filtrowanie i przetwarzanie dużych zbiorów danych.

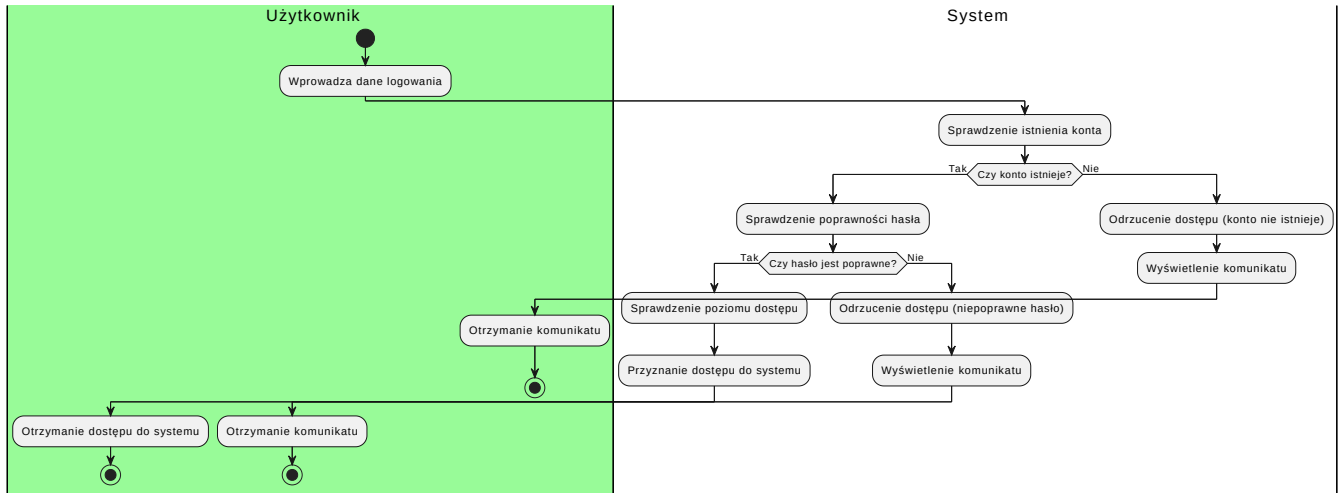
Kod diagramu

```
@startuml
|#palegreen|u| Użytkownik
|s| System

|u|
start
:Użytkownik wprowadza kryteria wyszukiwania;
|s|
:System filtruje dane na podstawie kryteriów;
|s|
:System prezentuje wyniki;
|u|
:Użytkownik otrzymuje wyniki;
stop
@enduml
```

(PB0005) Zarządzanie dostępem

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **zarządzaniem dostępem**. Proces ten składa się z kilku głównych etapów, w których uczestniczą dwie główne role:

1. Wprowadzenie danych logowania (User Inputs Credentials)

- Rola: Użytkownik (user)
- Opis: Użytkownik wprowadza swoje dane logowania .
- Kolor: #palegreen
- Oznaczenie: u

2. Sprawdzenie istnienia konta (Account Existence Check)

- Rola: System (system)
- Opis: System weryfikuje, czy konto użytkownika istnieje w bazie danych.
- Oznaczenie: s

3. Sprawdzenie poprawności hasła (Password Check)

- Rola: System (system)
- Opis: System sprawdza, czy wprowadzone hasło jest poprawne w stosunku do zapisanych danych.
- Oznaczenie: s

4. Sprawdzenie poziomu dostępu (Access Level Check)

- Rola: System (system)
- Opis: Jeśli dane logowania są poprawne, system sprawdza poziom dostępu użytkownika (np. admin, użytkownik standardowy).
- Oznaczenie: s

5. Przyznanie dostępu (Access Granted)

- Rola: System (system)
- Opis: System przyznaje dostęp do systemu użytkownikowi, jeśli dane logowania są poprawne i użytkownik ma odpowiednie uprawnienia.

- Oznaczenie: **S**

6. Odrzucenie dostępu (Access Denied)

- Rola: System (system)
- Opis: Jeśli konto nie istnieje, dane logowania są niepoprawne lub użytkownik nie ma odpowiednich uprawnień, dostęp zostaje odrzucony.
- Oznaczenie: **S**

7. Wyświetlenie komunikatu (Display Message)

- Rola: Użytkownik (user)
- Opis: Użytkownik otrzymuje komunikat o powodzeniu lub błędzie w zależności od wyniku logowania.
- Kolor: **#palegreen**
- Oznaczenie: **u**

Proces biznesowy

Proces biznesowy polega na weryfikacji danych użytkownika i przyznaniu dostępu do systemu. Ma na celu zapewnienie bezpieczeństwa i kontroli nad dostępem do zasobów systemu, zapobiegając nieautoryzowanemu dostępowi. Proces składa się z kilku kroków, w których użytkownik dostarcza dane, a system je weryfikuje, przyznając lub odrzucając dostęp.

Kod diagramu

```
@startuml
|#palegreen|u| Użytkownik
start
:Wprowadza dane logowania;

|s| System
:Sprawdzenie istnienia konta;
if (Czy konto istnieje?) then (Tak)
  :Sprawdzenie poprawności hasła;
  if (Czy hasło jest poprawne?) then (Tak)
    :Sprawdzenie poziomu dostępu;
    :Przyznanie dostępu do systemu;
  |u| Użytkownik
    :Otrzymanie dostępu do systemu;
    stop
  |s| System
  else (Nie)

    :Odrzucenie dostępu (niepoprawne hasło);

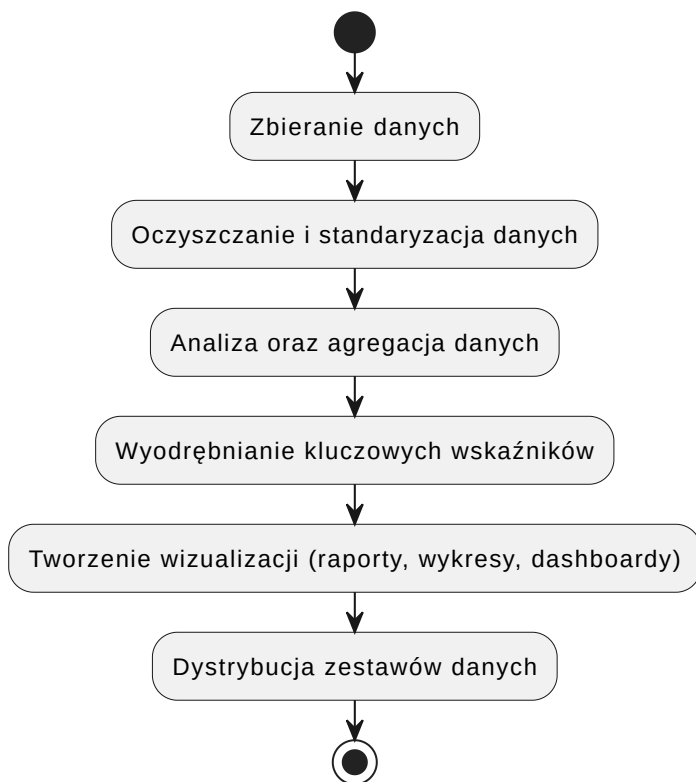
    :Wyświetlenie komunikatu;
  |u| Użytkownik
    :Otrzymanie komunikatu;
    stop
```

```
|s| System
endif
else (Nie)
:Odrzucenie dostępu (konto nie istnieje);

:Wyświetlenie komunikatu;
|u| Użytkownik
:Otrzymanie komunikatu;
stop
endif
@enduml
```

(PB0006) Prezentowanie zbiorów danych

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **prezentowaniem zbiorów danych**. Proces ten składa się z następujących etapów, w których uczestnicza jedna rola:

1. Zbieranie danych

- Rola: System (system)
- Opis: Dane są automatycznie pobierane z różnych źródeł przez system. Pracownik nie musi podejmować żadnych działań w tym zakresie.
- Oznaczenie: S

2. Oczyszczanie i standaryzacja danych

- Rola: System (system)
- Opis: System usuwa błędy, duplikaty oraz niekompletne rekordy. Dane są standaryzowane do jednolitego formatu zgodnego z wymaganiami analitycznymi.
- Oznaczenie: S

3. Analiza oraz agregacja danych:

- Rola: System (system)
- Opis: System analizuje dane w celu identyfikacji trendów, wzorców oraz anomalii. Możliwe jest grupowanie danych według określonych kategorii.
- Oznaczenie: S

4. Wyodrębnianie kluczowych wskaźników:

- Rola: System (system)
- Opis: Na podstawie analizy system generuje kluczowe wskaźniki, np. średnie wartości, odchylenia standardowe, rankingi.
- Oznaczenie: S

5. Tworzenie wizualizacji (raporty, wykresy, dashboardy):

- Rola: System (system)
- Opis: System generuje raporty, interaktywne wykresy i dashboardy w celu lepszej interpretacji danych. Dane mogą być prezentowane w tabelach, diagramach czy mapach ciepła.
- Oznaczenie: S

6. Dystrybucja zestawów danych

- Rola: System (system)
- Opis: System automatycznie udostępnia przetworzone dane użytkownikom poprzez aplikację webową, API lub eksport do plików (np. JSON, XML). Użytkownicy mogą pobierać raporty lub filtrować interesujące ich dane.
- Oznaczenie: S

Proces biznesowy polega na przekształceniu surowych danych w przejrzyste i wartościowe informacje, które mogą być używane do podejmowania decyzji, analizy trendów oraz monitorowania wskaźników wydajności.

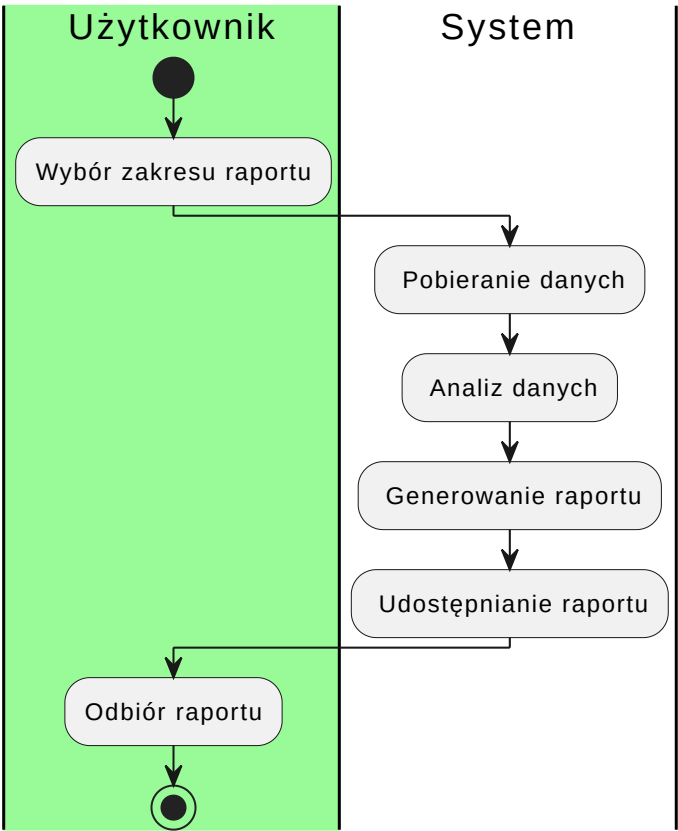
Kod diagramu

```
@startuml
    |s| System
    |s|
    start
    :Zbieranie danych;
```

```
:Oczyszczanie i standaryzacja danych;  
:Analiza oraz agregacja danych;  
:Wyodrębnianie kluczowych wskaźników;  
:Tworzenie wizualizacji (raporty, wykresy, dashboardy);  
:Dystrybucja zestawów danych;  
  
stop  
@enduml
```

(PB0007) Generowanie raportów o ruchu

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **generowaniem raportów o ruchu**:

1. Wybór zakresu raportu

- Rola: Użytkownik
- Opis: Użytkownik określa zakres danych, które mają być uwzględnione w raporcie, np. przedział czasowy, lokalizację, typy zdarzeń itp.
- Kolor: #palegreen
- Oznaczenie: u

2. Analiza danych

- Rola: System
- Opis: System przetwarza i analizuje dane dotyczące ruchu, agregując je, wykrywając anomalie oraz identyfikując trendy.

3. Generowanie raportu

- Rola: System
- Opis: System tworzy raport na podstawie zgromadzonych danych, który może zawierać wykresy, statystyki, mapy oraz inne wizualizacje ułatwiające interpretację wyników.

4. Udostępnienie raportu

- Rola: System
- Opis: Raport jest udostępniany użytkownikowi w wybranym formacie (np. PDF, CSV, JSON, interaktywna strona internetowa) i może być wysyłany na e-mail lub udostępniany przez API.

Proces biznesowy polega na automatycznym przetwarzaniu i analizie danych o ruchu, a następnie generowaniu raportów dostosowanych do potrzeb użytkownika. Dzięki temu organizacje mogą podejmować świadome decyzje oparte na rzeczywistych danych, optymalizować procesy i reagować na zmiany w ruchu w czasie rzeczywistym.

Kod diagramu

```
@startuml
|#palegreen|u|Użytkownik
|System|

|u|
start
:Wybór zakresu raportu;

|System|
: Pobieranie danych;

: Analiz danych;

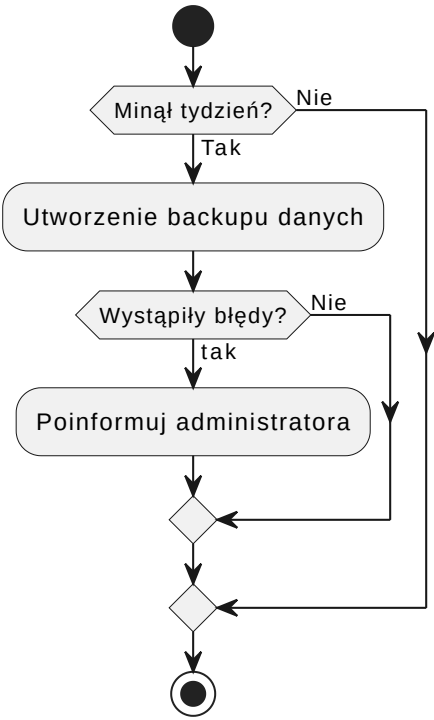
: Generowanie raportu;

: Udostępnianie raportu;

|u|
:Odbiór raportu;

stop
@enduml
```

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **regularną archiwizacją danych** :

1. Cykliczna aktywacja

- Rola: System
- System co tydzień aktywuje proces archiwizacji danych.

2. Utworzenie backupu danych

- Rola: System
- System tworzy kopię zapasową odzwierciedlającą aktualny stan bazy danych.

3. Poinformowanie administratora

- Rola: System
- System w przypadku wykrycia błędów podczas tworzenia kopii wysyła o tym wiadomość do administratora.

Proces biznesowy polega na regularnym tworzeniu kopii zapasowej aktualnego stanu danych w systemie. Celem tego procesu jest posiadanie możliwości cofnięcia się do poprzedniej wersji danych w przypadku poważniejszego błędu.

Kod diagramu

```

@startuml
start

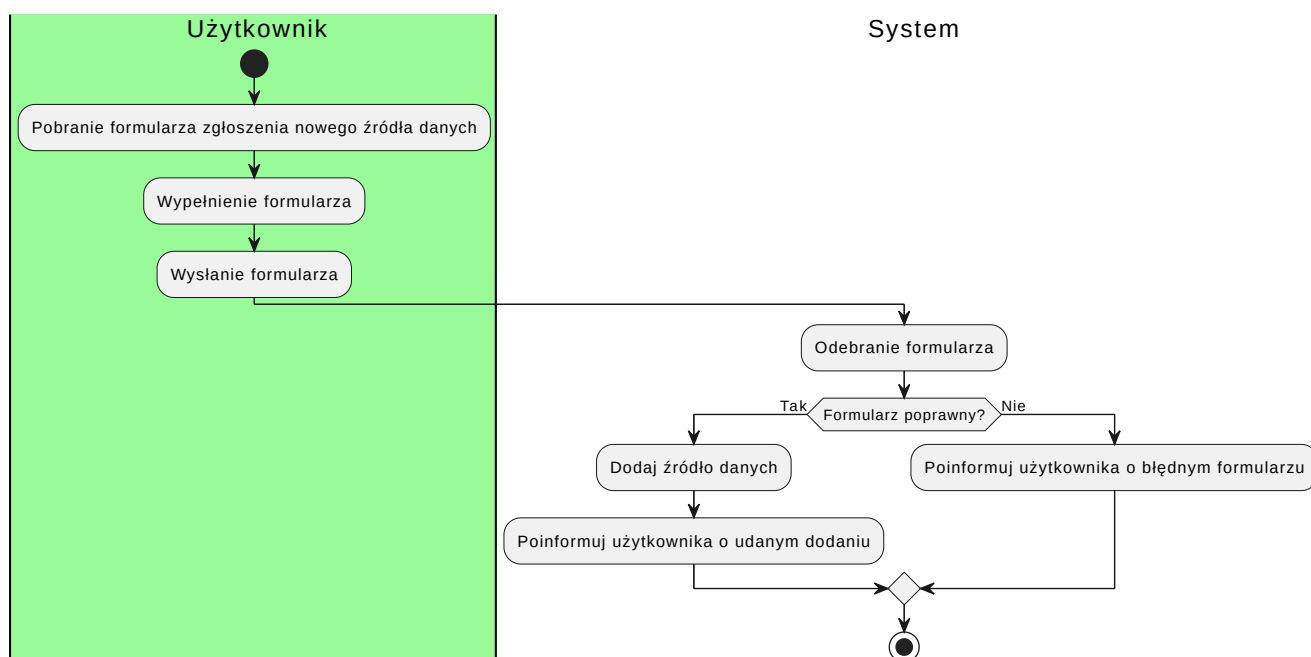
if (Minął tydzień?) then (Tak)
  :Utworzenie backupu danych;
  if (Wystąpiły błędy?) then (tak)
    :Poinformuj administratora;
  else (Nie)
  endif
else (Nie)
endif

stop
@enduml

```

(PB0009) Zgłaszanie nowych źródeł danych

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany ze **zgłaszaniem nowych źródeł danych** :

1. Pobranie formularza

- Rola: Użytkownik
- Formularz zgłoszeniowy jest dostępny publicznie do pobrania.

2. Wypełnienie formularza

- Rola: Użytkownik

- Formularz należy uzupełnić odpowiednimi danymi które następnie będą walidowane

3. Walidacja formularza

- Rola: System
- System sprawdza poprawność formularza i w zależności od wyniku dodaje źródło danych lub informuje klienta o niepoprawnym formularzu.

Proces biznesowy polega na dodawaniu nowych źródeł danych za pomocą formularza. Celem tego procesu jest posiadanie możliwości szybkiego i sprawnego poszerzania naszej bazy danych.

Kod diagramu

```
@startuml
    actor Uzytkownik as Użytkownik
    participant System
    start
        :Pobranie formularza zgłoszenia nowego źródła danych;
        :Wypełnienie formularza;
        :Wysłanie formularza;
    and
        :Odebranie formularza;
    if (Formularz poprawny?) then (Tak)
        :Dodaj źródło danych;
        :Poinformuj użytkownika o udanym dodaniu;
    else (Nie)
        :Poinformuj użytkownika o błędnym formularzu;
    endif
    stop
@enduml
```

(PB0010) Ocenianie zbiorów danych pod kątem przydatności

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **Ocenianiem zbiorów danych pod kątem przydatności**:

1. Wystawienie oceny

- Rola: Użytkownik
- Użytkownicy naszego serwisu mają możliwość oceniania źródeł danych pod kątem przydatności w 5 stopniowej skali (gwiazdki).

2. Weryfikacja unikalności

- Rola: System
- System weryfikuje czy użytkownik nie ocenił już danego zbioru danych.

3. Zaktualizowanie oceny

- Rola: System
- Wyświetlana ocena danego źródła jest średnią wszystkich poprzednich ocen oraz nowej dodanej.

Proces biznesowy polega na ocenieniu źródła danych. Celem tego procesu jest ułatwienie innym użytkownikom wyszukiwania wartościowych zbiorów danych.

Kod diagramu

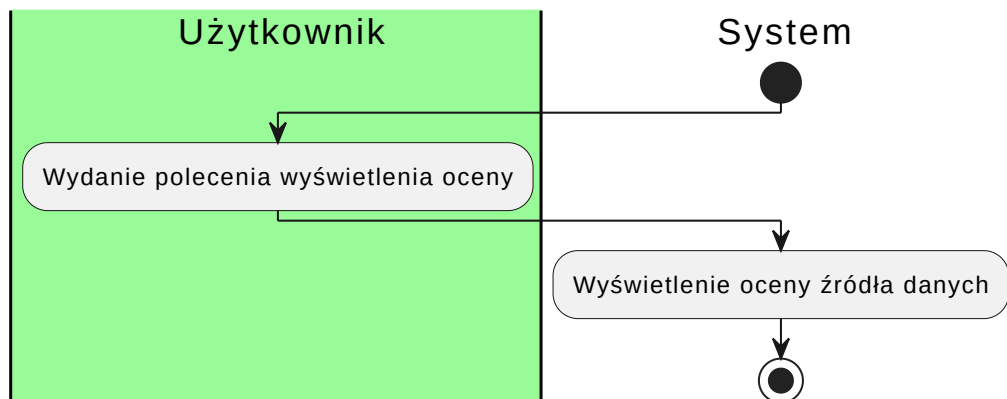
```
@startuml
| #palegreen | u | Użytkownik
| s | System
| u |
start
: Wystawienie oceny dla danego źródła danych;
| s |
: Weryfikacja unikalności oceny;
```

```
:Zaktualizowanie średniej ocen o nową ocenę;
```

```
stop
@enduml
```

(PB0011) Prezentowanie ocen danych

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **prezentowaniem ocen danych** :

1. Wysłanie polecenia wyświetlenia oceny

- Rola: Użytkownik
- Użytkownik wyraża chęć zobaczenia oceny danego źródła danych.

2. Wyświetlenie oceny źródła danych

- Rola: System
- Każde źródło danych jest opisane swoją oceną, która jest średnią ocen wystawionych przez użytkowników w 5-stopniowej skali (gwiazdki).

Proces biznesowy polega na dodawaniu prezentowaniu oceny źródła danych. Celem tego procesu jest ułatwienie klientom rozpoznawania wartościowych zbiorów danych.

Kod diagramu

```
@startuml

|#palegreen|u| Użytkownik
|s| System

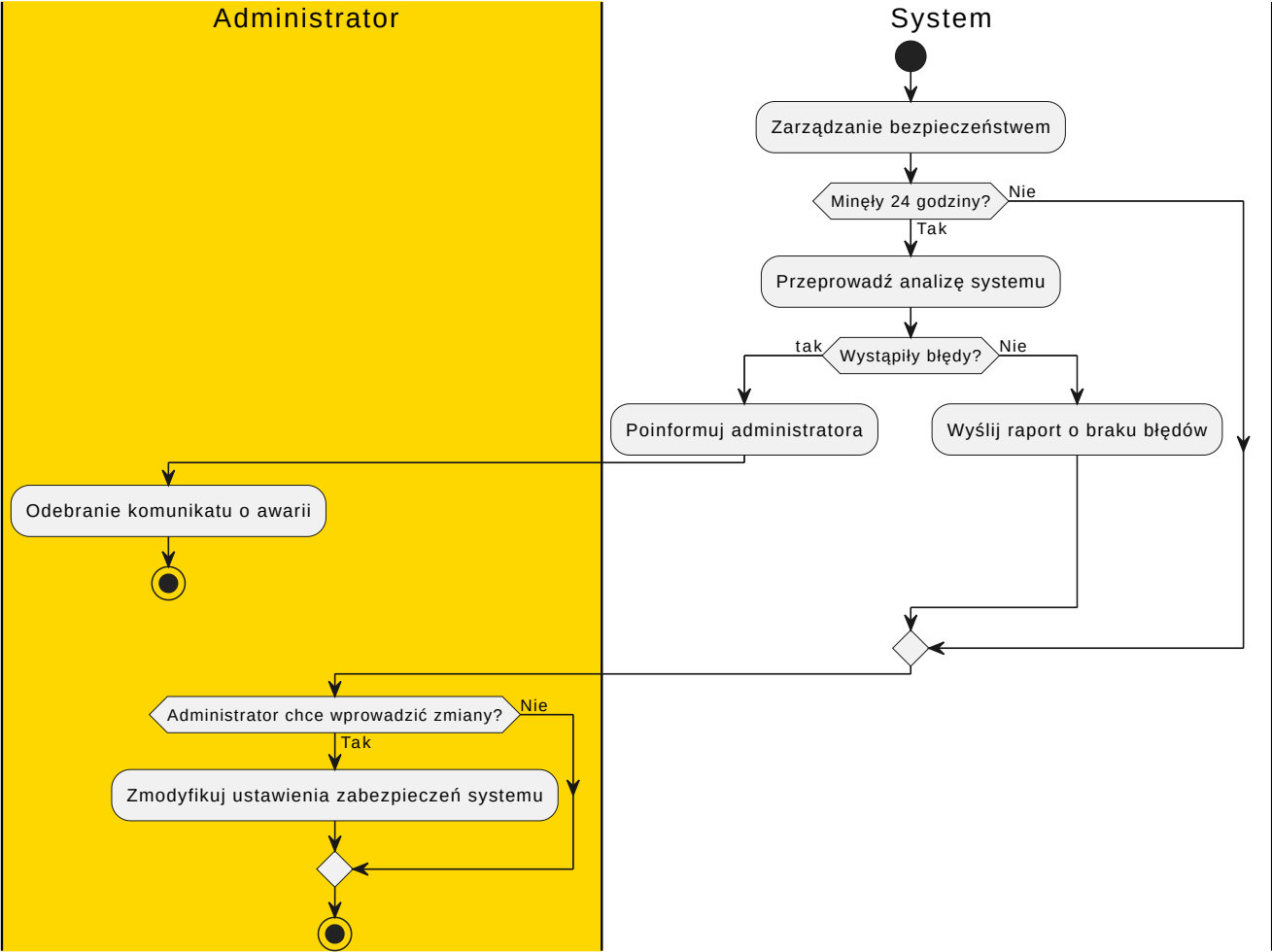
start
|u|
```

```
:Wydanie polecenia wyświetlenia oceny;
|s|
:Wyświetlenie oceny źródła danych;

stop
@enduml
```

(PB0012) Zarządzanie bezpieczeństwem

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany ze **zarządzaniem bezpieczeństwem** :

1. Generowanie raportu

- Rola: System
- System regularnie generuje raport o stanie.

2. Poinformowanie administratora

- Rola: System
- Administrator zostaje poinformowany o potencjalnym zagrożeniu i następnie podejmuje decyzje odnośnie kolejnych działań.

3. Modyfikowanie ustawień bezpieczeństwa

- Rola: Administrator
- Administrator może włączyć lub wyłączyć dane zabezpieczenia, jeżeli jest taka potrzeba.

Proces biznesowy polega na zarządzaniu bezpieczeństwem. Celem tego procesu jest odpowiednie reagowanie na sytuacje zagrożenia.

Kod diagramu

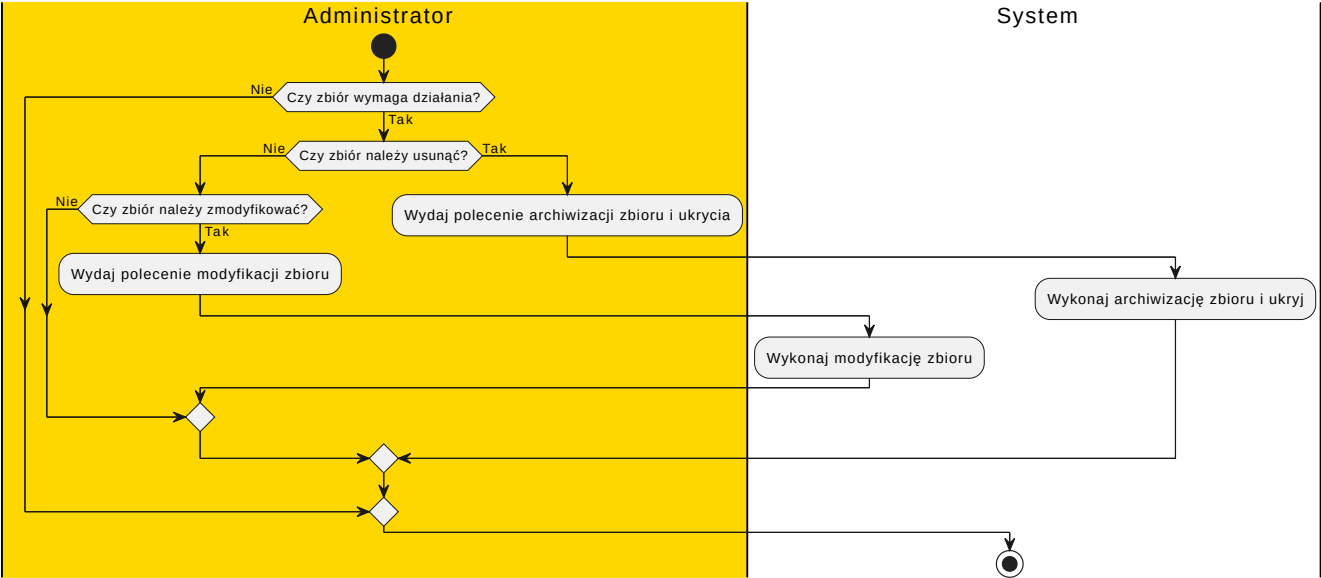
```
@startuml
|#gold|a| Administrator
|s| System
start
:Zarządzanie bezpieczeństwem;

if (Minęły 24 godziny?) then (Tak)
:Przeprowadź analizę systemu;
if (Wystąpiły błędy?) then (tak)
:Poinformuj administratora;
|a|
:Odebranie komunikatu o awarii;
stop
else (Nie)
|s|
:Wyślij raport o braku błędów;
endif
else (Nie)
endif
|a|
if (Administrator chce wprowadzić zmiany?) then (Tak)
:Zmodyfikuj ustawienia zabezpieczeń systemu;
else (Nie)
endif

stop
@enduml
```

(PB0013) Zarządzanie zbiorami danych

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **zarządzaniem zbiorami danych** :

1. Archiwizowanie zbioru

- o Rola: System
- o System dokonuje archiwizacji zbioru i ukrywa go.

2. Modyfikowanie zbioru

- o Rola: System
- o System realizuje odpowiednią modyfikację zbioru.

Proces biznesowy polega na zarządzaniu zbiorami danych. Celem tego procesu jest posiadanie spójnej i możliwie lekkiej bazy danych.

Kod diagramu

```
@startuml
|#gold|a| Administrator
|s| System
|a|

start
if (Czy zbiór wymaga działania?) then (Tak)
    if (Czy zbiór należy usunąć?) then (Nie)
        if (Czy zbiór należy zmodyfikować?) then (Nie)
        else (Tak)
            :Wydaj polecenie modyfikacji zbioru;
            |s|
            :Wykonaj modyfikację zbioru;
        endif
    else (Tak)
        :Wydaj polecenie archiwizacji zbioru i ukrycia;
    end
endif
if (Czy są jeszcze zbiory do przetworzenia?) then (Tak)
    goto start
else (Nie)
    stop
endif
```

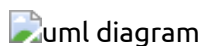
```
    else (Tak)
    |a|
    :Wydaj polecenie archiwizacji zbioru i ukrycia;
    |s|
    :Wykonaj archiwizację zbioru i ukryj;
    endif
else (Nie)
endif

stop

@enduml
```

(PB0014) Informowanie o awariach

Diagram



Opis

Na podstawie diagramu PlantUML, można opisać proces biznesowy związany z **informowaniem o awariach** :

1. Odebranie komunikatu o awarii

- Rola: Administrator
- Administrator odbiera na swoim urządzeniu komunikat o awarii.

2. Przeanalizowanie awarii

- Rola: Administrator
- Administrator analizuje dostarczony mu ogólny raport o awarii.

3. Naprawienie problemu

- Rola: Administrator
- Jeżeli znana jest przyczyna awarii, administrator podejmuje odpowiednie działania w celu naprawienia błędu

4. Generowanie szczegółowego raportu

- Rola: System
- Jeżeli nie jest znana przyczyna awarii, system jest w stanie wygenerować szczegółowy raport dotyczący awarii

5. Przenalizowanie szczegółowego raportu

- Rola: Administrator
- Administrator analizuje szczegółowy raport dotyczący awarii

6. Przywrócenie wcześniejszej wersji systemu

- Rola: Administrator
- Administrator przywraca poprzednią, działającą wersję systemu.

7. Podjęcie odpowiednich kroków

- Rola: Administrator
- Jeżeli wciąż nie jest znana przyczyna, administrator może podjąć odpowiednie kroki jak np. poinformowanie o tym innych osób, wyłączenie niedziałającego modułu.

Proces biznesowy polega na informowaniu o awariach systemu. Celem tego procesu jest możliwie szybkie informowanie o awariach w celu efektywnej odpowiedzi na zaistniały problem.

Kod diagramu

```
@startuml
|#gold|a| Administrator
|s| System
|a|
start
:Odbierz komunikat o awarii;
:Przeanalizuj awarię;
if (Znana jest przyczyna awarii?) then (Tak)
    :Napraw problem;
else (Nie)
    :Wydadź polecenie generowania szczegółowego raportu awarii;
    |s|
    :Generuj szczegółowy raport awarii;
    |a|
    :Analizuj szczegółowy raport;
    if (Znana jest przyczyna awarii?) then (Tak)
        :Napraw problem;
    else (Nie)
        :Przywróć wcześniejszą wersję systemu;
        :Podejmij odpowiednie kroki;
    endif
endif
stop
@enduml
```

3. wizja systemu

Motto

"Otwarte dane, nieograniczone możliwości" "Łączymy świat z danymi publicznymi - prosto, bezpiecznie i efektywnie" "Jednolity dostęp do wiarygodnych danych publicznych dla wszystkich, zawsze i wszędzie"

3.1 Opisy problemów

Problemy

W tej sekcji przedstawiono kluczowe problemy, które system Zarządzania Otwartymi Danymi Publicznymi ma rozwiązać.

1. Rozproszenie zbiorów danych publicznych

Opis problemu: Obecnie dane publiczne są rozproszone w różnych systemach i platformach, co utrudnia ich odnalezienie i wykorzystanie.

Wpływ na interesariuszy:

- Klienci/Użytkownicy danych muszą przeszukiwać wiele źródeł, co jest czasochłonne i nieefektywne
- Programiści API nie mogą łatwo integrować swoich aplikacji z wieloma różnymi systemami
- Researcherzy/Analitycy danych mają problem z kompleksowym dostępem do potrzebnych zbiorów

Potrzeby wynikające z problemu:

- Centralne repozytorium danych publicznych
- Ujednolicony interfejs dostępu
- Spójny system metadanych opisujących zbiory

2. Niska jakość i nieaktualność danych

Opis problemu: Dostępne zbiory danych często zawierają błędy, niespójności lub są nieaktualne, co obniża ich wartość i przydatność.

Wpływ na interesariuszy:

- Klienci/Użytkownicy danych nie mogą polegać na otrzymanych informacjach
- Decydenci otrzymują niepełne lub błędne informacje utrudniające podejmowanie decyzji
- Dostawcy danych tracą wiarygodność

Potrzeby wynikające z problemu:

- Mechanizmy weryfikacji poprawności danych
- System regularnych aktualizacji
- Narzędzia do wykrywania anomalii i niespójności
- Możliwość zgłaszania błędów przez użytkowników

3. Trudności w zarządzaniu zbiorami danych

Opis problemu: Zarządzanie dużymi zbiorami danych jest skomplikowane, szczególnie gdy pochodzą z różnych źródeł i mają różne formaty.

Wpływ na interesariuszy:

- Administratorzy systemu zmagają się z chaotycznym systemem organizacji danych
- Dostawcy danych muszą ręcznie konwertować dane do odpowiednich formatów
- Archiwiści danych mają problem z utrzymaniem spójności i wersjonowaniem zbiorów

Potrzeby wynikające z problemu:

- Efektywne narzędzia do zarządzania metadanymi
- Automatyzacja procesów importu i eksportu danych
- System zarządzania schematami danych

4. Problemy z bezpieczeństwem i kontrolą dostępu

Opis problemu: Brak odpowiednich mechanizmów zabezpieczeń oraz kontroli dostępu do wrażliwych danych.

Wpływ na interesariuszy:

- Audytorzy/Inspektorzy Ochrony Danych zgłaszają niezgodność z przepisami
- Kierownictwo instytucji obawia się naruszeń bezpieczeństwa i konsekwencji prawnych
- Użytkownicy nie mają pewności, czy ich dane osobowe są odpowiednio chronione

Potrzeby wynikające z problemu:

- Mechanizmy uwierzytelniania i autoryzacji
- Szyfrowanie wrażliwych danych
- System śledzenia dostępu i zmian

5. Nieefektywne wyszukiwanie i filtrowanie danych

Opis problemu: Istniejące systemy nie zapewniają zaawansowanych możliwości wyszukiwania i filtrowania danych.

Wpływ na interesariuszy:

- Klienci/Użytkownicy danych tracą czas na ręczne przeszukiwanie zbiorów
- Researcherzy/Analitycy danych nie mogą szybko dotrzeć do potrzebnych informacji
- Programiści API mają ograniczone możliwości przeszukiwania danych przez API

Potrzeby wynikające z problemu:

- Zaawansowane mechanizmy wyszukiwania i filtrowania
- Indeksowanie danych zwiększające wydajność wyszukiwania
- Intuicyjny interfejs wyszukiwania

6. Brak integracji z istniejącymi systemami

Opis problemu: Trudności w integracji nowego systemu z istniejącą infrastrukturą i systemami zewnętrznymi.

Wpływ na interesariuszy:

- Dział IT musi zarządzać wieloma niepowiązanymi systemami
- Administratorzy systemu wykonują podwójną pracę, wprowadzając te same dane do różnych systemów
- Dostawcy danych muszą dublować operacje w różnych systemach

Potrzeby wynikające z problemu:

- Integracja z platformą CKAN
- Dobrze udokumentowane API
- Mechanizmy synchronizacji danych

7. Ograniczone możliwości prezentacji i wizualizacji danych

Opis problemu: Dane są często prezentowane w surowej formie, bez odpowiednich narzędzi do ich wizualizacji i analizy.

Wpływ na interesariuszy:

- Klienci/Użytkownicy danych mają trudności ze zrozumieniem złożonych zbiorów
- Decydenci nie mogą szybko uzyskać potrzebnych informacji w przystępnej formie
- Researcherzy/Analitycy danych muszą używać zewnętrznych narzędzi do wizualizacji

Potrzeby wynikające z problemu:

- Narzędzia do wizualizacji danych
- Generowanie raportów w różnych formatach
- Prezentacja danych w formie wykresów i tabel

8. Wysokie koszty utrzymania systemów danych

Opis problemu: Utrzymanie wielu rozproszonych systemów generuje wysokie koszty operacyjne i wymaga dużych zasobów IT.

Wpływ na interesariuszy:

- Kierownictwo instytucji zмага się z rosnącymi kosztami IT
- Dział IT musi dzielić zasoby między wiele systemów
- Zespół deweloperski spędza czas na utrzymaniu zamiast na rozwoju

Potrzeby wynikające z problemu:

- Optymalizacja wykorzystania zasobów
- Automatyzacja rutynowych operacji
- Skalowalność systemu

9. Brak mechanizmów do zbierania informacji zwrotnej

Opis problemu: Istniejące systemy nie umożliwiają użytkownikom oceniania i komentowania zbiorów danych.

Wpływ na interesariuszy:

- Dostawcy danych nie otrzymują informacji o użyteczności udostępnianych zbiorów
- Klienci/Użytkownicy danych nie mogą dzielić się spostrzeżeniami i wskazywać problemów
- Kierownictwo instytucji nie ma informacji zwrotnej o wartości udostępnianych danych

Potrzeby wynikające z problemu:

- System ocen i komentarzy
- Mechanizmy zbierania i analizy opinii użytkowników
- Możliwość zgłaszania błędów w danych

10. Niedostosowanie do potrzeb osób z niepełnosprawnościami

Opis problemu: Obecne systemy często nie spełniają standardów dostępności dla osób z niepełnosprawnościami.

Wpływ na interesariuszy:

- Klienci/Użytkownicy z niepełnosprawnościami są wykluczeni z dostępu do danych publicznych
- Audytorzy/Inspektorzy zgłaszają niezgodność z regulacjami dotyczącymi dostępności
- Kierownictwo instytucji narażone jest na konsekwencje prawne

Potrzeby wynikające z problemu:

- Interfejs zgodny ze standardami WCAG
- Kompatybilność z technologiami wspomagającymi
- Alternatywne metody dostępu do danych

3.2 Interesariusze

Interesariusze systemu

W tej sekcji przedstawiono osoby i grupy, które są bezpośrednio lub pośrednio zainteresowane powstaniem systemu Zarządzania Otwartymi Danymi Publicznymi i na które wdrożenie tego systemu będzie miało wpływ.

1. Administrator systemu

Typ interesariusza: Użytkownik wewnętrzny, personel techniczny

Opis: Osoby odpowiedzialne za zarządzanie systemem, rejestrowanie nowych źródeł danych, otrzymywanie powiadomień o błędach oraz tworzenie kopii zapasowych.

Zakres obowiązków:

- Zarządzanie konfiguracją systemu i zbiorami danych
- Monitorowanie wydajności i dostępności
- Zarządzanie kontami użytkowników i uprawnieniami
- Rozwiązywanie problemów technicznych

Kryteria zadowolenia:

- Intuicyjny panel administracyjny
- Narzędzia do automatycznego monitorowania
- Efektywne zarządzanie użytkownikami

Zadania w projekcie:

- Konsultowanie wymagań technicznych
- Testowanie funkcji administracyjnych
- Przygotowanie procedur backupu

2. Dostawca danych

Typ interesariusza: Użytkownik wewnętrzny, ekspert dziedzinowy

Opis: Pracownicy instytucji dostarczający dane do systemu, posiadający wiedzę ekspercką o tych danych.

Zakres obowiązków:

- Przygotowywanie danych do publikacji
- Weryfikacja poprawności danych
- Aktualizowanie zbiorów danych
- Zarządzanie metadanymi i schematami

Kryteria zadowolenia:

- Łatwy proces dodawania danych
- Automatyczna walidacja zgodności
- Wsparcie dla różnych formatów

Zadania w projekcie:

- Definiowanie wymagań dotyczących formatów
- Testowanie funkcji zarządzania danymi
- Dostarczanie przykładowych zbiorów

3. Klient/Użytkownik danych

Typ interesariusza: Użytkownik końcowy

Opis: Osoby korzystające z systemu w celu uzyskania informacji publicznych (obywatele, dziennikarze, aktywiści, przedsiębiorcy).

Zakres obowiązków:

- Wyszukiwanie i przeglądanie danych
- Ocenianie wartości zbiorów danych
- Zgłaszanie problemów z danymi

Kryteria zadowolenia:

- Intuicyjny interfejs użytkownika
- Łatwość wyszukiwania i filtrowania
- Czytelna prezentacja w różnych formatach
- Dostępność na różnych urządzeniach

Zadania w projekcie:

- Udział w badaniach użyteczności

- Testowanie interfejsu
- Zgłaszanie sugestii

4. Programista API

Typ interesariusza: Klient zewnętrzny, użytkownik techniczny

Opis: Deweloperzy tworzący aplikacje wykorzystujące dane z systemu poprzez API.

Zakres obowiązków:

- Integracja swoich aplikacji z API systemu
- Tworzenie usług bazujących na danych publicznych

Kryteria zadowolenia:

- Dobrze udokumentowane API
- Stabilność i wydajność interfejsu
- Narzędzia do testowania

Zadania w projekcie:

- Konsultowanie specyfikacji API
- Testowanie prototypów

5. Researcher / Analityk danych

Typ interesariusza: Użytkownik zewnętrzny

Opis: Naukowcy, badacze i analitycy wykorzystujący dane do celów badawczych i edukacyjnych.

Zakres obowiązków:

- Analiza i interpretacja danych publicznych
- Tworzenie modeli i wizualizacji
- Publikowanie wyników badań

Kryteria zadowolenia:

- Dostęp do kompletnych zbiorów
- Zaawansowane opcje wyszukiwania
- Możliwość pobierania w formatach analitycznych

Zadania w projekcie:

- Definiowanie wymagań analitycznych
- Ocena przydatności danych do badań

6. Kierownictwo instytucji / Decydenci

Typ interesariusza: Zamawiający, osoba decyzyjna

Opis: Kadra kierownicza odpowiedzialna za strategiczne decyzje i odbierająca dane wyżej w hierarchii.

Zakres obowiązków:

- Decyzje strategiczne o zakresie danych
- Zatwierdzanie budżetu
- Nadzór nad zgodnością z regulacjami

Kryteria zadowolenia:

- Zgodność z przepisami
- Pozytywny wpływ wizerunkowy
- Mierzalne korzyści z udostępniania danych

Zadania w projekcie:

- Zatwierdzanie specyfikacji
- Zapewnienie zasobów
- Ustalanie priorytetów

7. Zespół deweloperski

Typ interesariusza: Personel zaangażowany w tworzenie systemu

Opis: Programiści i projektanci pracujący nad budową systemu.

Zakres obowiązków:

- Implementacja funkcjonalności
- Testowanie i debugowanie
- Integracja z zewnętrznymi systemami

Kryteria zadowolenia:

- Jasne wymagania
- Dostęp do zasobów
- Możliwość wykorzystania nowych technologii

Zadania w projekcie:

- Implementacja funkcji
- Projektowanie interfejsu
- Dokumentacja techniczna

8. Dział IT instytucji

Typ interesariusza: Użytkownik wewnętrzny, personel techniczny

Opis: Zespół odpowiedzialny za integrację systemu z istniejącą infrastrukturą.

Zakres obowiązków:

- Zapewnienie infrastruktury
- Integracja z systemami instytucji
- Wsparcie techniczne pierwszej linii

Kryteria zadowolenia:

- Kompatybilność z infrastrukturą
- Łatwość utrzymania
- Solidna dokumentacja

Zadania w projekcie:

- Konsultowanie wymagań infrastrukturalnych
- Przygotowanie środowiska

9. Audytor / Inspektor Ochrony Danych

Typ interesariusza: Specjalista ds. zgodności

Opis: Osoby weryfikujące zgodność systemu z przepisami prawa i standardami bezpieczeństwa.

Zakres obowiązków:

- Ocena zgodności z RODO, NIS2
- Monitorowanie procesów przetwarzania
- Przeprowadzanie audytów bezpieczeństwa

Kryteria zadowolenia:

- Przejrzysta dokumentacja procesów
- Mechanizmy zapewniające zgodność
- Ślad audytowy operacji

Zadania w projekcie:

- Weryfikacja zgodności rozwiązań
- Ocena wpływu na ochronę danych

10. Archiwista danych

Typ interesariusza: Użytkownik wewnętrzny

Opis: Osoby odpowiedzialne za długoterminowe przechowywanie i wersjonowanie danych.

Zakres obowiązków:

- Zarządzanie archiwizacją danych
- Utrzymywanie historii danych
- Zarządzanie politykami retencji

Kryteria zadowolenia:

- Skuteczne mechanizmy archiwizacji
- Możliwość odtworzenia historycznych wersji
- Efektywne zarządzanie przestrzenią

Zadania w projekcie:

- Definiowanie strategii archiwizacji
- Testowanie mechanizmów wersjonowania

11. Reklamodawcy

Typ interesariusza: Partner biznesowy

Opis: Firmy zainteresowane dotarciem do odbiorców systemu poprzez reklamy.

Zakres obowiązków:

- Dostarczanie materiałów reklamowych
- Opłacanie usług reklamowych

Kryteria zadowolenia:

- Przejrzyste raportowanie skuteczności
- Dotarcie do właściwej grupy docelowej
- Pozytywny zwrot z inwestycji

Zadania w projekcie:

- Konsultacja modelu monetyzacji
- Testowanie platformy reklamowej

3.3 Cechy funkcjonalne

Cechy funkcjonalne systemu

System Zarządzania Otwartymi Danymi Publicznymi powinien realizować następujące funkcjonalności, pogrupowane według stopnia ważności.

Priorytet wysoki

1. Zarządzanie metadanymi udostępnionych zbiorów danych

- System musi umożliwiać przechowywanie i zarządzanie metadanymi opisującymi zbiory danych.
- Metadane powinny zawierać informacje o źródle danych, dacie utworzenia, aktualizacji, formacie, itp.

2. Zarządzanie dystrybucjami zbiorów danych i ich powiązanie ze źródłami danych

- System powinien umożliwiać zarządzanie różnymi dystrybucjami tych samych zbiorów danych.
- Musi istnieć możliwość śledzenia pochodzenia danych i powiązania ich ze źródłami.

3. Możliwość wygodnego wprowadzania, przeglądania, filtrowania i wyszukiwania zbiorów danych i ich dystrybucji

- System powinien zapewniać intuicyjny interfejs do wprowadzania nowych danych.
- Użytkownicy muszą mieć możliwość efektywnego przeszukiwania i filtrowania zbiorów danych.

4. Zarządzanie schematami danych i przypisywanie schematów danych do zbiorów danych

- System musi umożliwiać definiowanie schematów danych.
- Użytkownik powinien móc przypisywać schematy do odpowiednich zbiorów danych.

5. Weryfikacja zgodności dystrybucji zbiorów danych ze schematami

- System powinien automatycznie weryfikować, czy dane są zgodne z przypisanymi schematami.
- W przypadku wykrycia niezgodności, system musi informować o problemach.

6. Import i eksport danych w różnych formatach z zachowaniem zgodności ze schematami

- System musi obsługiwać import danych z różnych źródeł i w różnych formatach.
- Eksport danych powinien być możliwy w wielu formatach, z zachowaniem zgodności ze schematami.

7. Integracja z systemem CKAN

- System powinien integrować się z platformą CKAN do zarządzania danymi.
- Integracja powinna umożliwiać wymianę danych między systemami.

8. Filtrowanie i wyszukiwanie danych

- Użytkownicy powinni móc filtrować dane według różnych kryteriów.
- System musi zapewniać zaawansowane mechanizmy wyszukiwania.

9. Autentykacja danych

- System musi weryfikować autentyczność danych.
- Powinien istnieć mechanizm weryfikacji źródła danych.

10. Autoryzacja użytkowników

- System musi posiadać mechanizm zarządzania uprawnieniami użytkowników.
- Różni użytkownicy powinni mieć różne poziomy dostępu.

11. Przyznawanie dostępu

- Administrator powinien móc przyznawać i odbierać dostęp do systemu.
- System musi umożliwiać definiowanie ról i uprawnień.

12. Sprawdzenie poziomu dostępu użytkownika

- System musi weryfikować uprawnienia użytkownika przed wykonaniem operacji.
- Poziomy dostępu powinny być sprawdzane w czasie rzeczywistym.

13. Dodawanie danych przez API

- System musi udostępniać API do automatycznego dodawania danych.
- API powinno być dobrze udokumentowane i zabezpieczone.

14. Prezentowanie wszystkich zbiorów danych

- System musi prezentować listę wszystkich dostępnych zbiorów danych.
- Prezentacja powinna być przejrzysta i umożliwiać sortowanie.

15. Ulepszanie jakości danych

- System powinien posiadać mechanizmy do poprawy jakości przechowywanych danych.
- Należy implementować procesy weryfikacji i czyszczenia danych.

16. Utrzymanie aktualności danych

- System musi zapewniać mechanizmy regularnego aktualizowania danych.
- Użytkownicy powinni być informowani o statusie aktualności danych.

17. Wykrywanie anomalii

- System powinien automatycznie wykrywać anomalie i niespójności w danych.
- O wykrytych anomaliiach system powinien informować odpowiedzialnych użytkowników.

18. Zapewnienie bezpieczeństwa danych

- System musi implementować mechanizmy zabezpieczające dane przed nieautoryzowanym dostępem.
- Należy stosować odpowiednie mechanizmy kryptograficzne i kontroli dostępu.

19. Wykrywanie błędów w zbiorach

- System powinien automatycznie wykrywać błędy w zbiorach danych.
- Wykryte błędy powinny być raportowane i oznaczane do korekty.

20. Szyfrowanie danych wrażliwych

- Dane wrażliwe muszą być przechowywane w formie zaszyfrowanej.
- System powinien automatycznie rozpoznawać i szyfrować wrażliwe informacje.

21. Szyfrowanie przesyłu

- Komunikacja z systemem musi być szyfrowana.
- Należy stosować aktualne protokoły kryptograficzne do zabezpieczenia transmisji.

22. Automatyczna cykliczna walidacja zbiorów

- System powinien cyklicznie przeprowadzać walidację zbiorów danych.
- Wyniki walidacji powinny być raportowane odpowiednim użytkownikom.

Priorytet średni

1. System ocen oraz uwag od użytkowników

- Użytkownicy powinni móc oceniać zbiory danych i pozostawiać komentarze.
- System powinien agregować i wyświetlać oceny.

2. Prezentacja wyników wyszukiwania w różnych formatach

- System powinien prezentować wyniki wyszukiwania w różnych formatach (tabele, wykresy).
- Użytkownik powinien móc wybierać sposób prezentacji.

3. Wyodrębnianie kluczowych wskaźników

- System powinien automatycznie analizować dane i wyodrębniać kluczowe wskaźniki.
- Wskaźniki powinny być prezentowane w czytelny sposób.

4. Zgłaszanie błędów danych

- Użytkownicy powinni móc zgłaszać błędy w danych.
- System powinien zarządzać zgłoszeniami i umożliwiać ich weryfikację.

5. Generowanie raportów

- System musi umożliwiać generowanie raportów na podstawie danych.
- Raporty powinny być dostępne w różnych formatach.

6. Przedstawienie historii zmian w przystępny sposób

- System powinien śledzić historię zmian w danych.
- Historia zmian powinna być prezentowana w sposób czytelny dla użytkownika.

7. Powiadomienia o aktualizacjach

- System powinien informować użytkowników o aktualizacjach zbiorów danych.
- Użytkownicy powinni móc subskrybować powiadomienia dla określonych zbiorów.

8. Integracja narzędzi do wizualizacji danych

- System powinien integrować narzędzia umożliwiające wizualizację danych.
- Wizualizacje powinny być interaktywne i konfigurowalne.

9. Aktualizowanie wybranych zbiorów co sekundę

- System powinien umożliwiać częste aktualizacje wybranych zbiorów danych.
- Mechanizm aktualizacji powinien być wydajny i nie obciążać systemu.

10. Udostępnianie danych wyżej w hierarchii

- System powinien umożliwiać udostępnianie wybranych zbiorów danych na wyższe poziomy hierarchii.
- Proces udostępniania powinien być kontrolowany i zabezpieczony.

11. Archiwizacja danych

- System powinien automatycznie archiwizować starsze dane.
- Dane archiwalne powinny być dostępne na żądanie.

12. Wersjonowanie danych

- System musi umożliwiać przechowywanie różnych wersji tych samych zbiorów danych.
- Użytkownicy powinni mieć możliwość porównania różnych wersji danych.

13. Przechowywanie logów systemu w celu audytu

- System powinien przechowywać logi wszystkich operacji.
- Logi powinny być dostępne na potrzeby audytu i analizy bezpieczeństwa.

14. Zbieranie i analizowanie ocen użytkowników

- System powinien zbierać opinie użytkowników o danych i funkcjonalnościach.
- Zebrane opinie powinny być analizowane w celu poprawy jakości systemu.

15. Automatyzacja aktualizacji

- System powinien umożliwiać automatyczne aktualizacje danych z zewnętrznych źródeł.
- Proces aktualizacji powinien być monitorowany i raportowany.

16. Analiza wykorzystania danych

- System powinien analizować, które zbiory danych są najczęściej używane.
- Analiza powinna dostarczać informacji o wzorcach korzystania z danych.

17. Wykrywanie zagrożeń bezpieczeństwa

- System powinien monitorować próby nieautoryzowanego dostępu.
- O potencjalnych zagrożeniach powinni być informowani administratorzy.

18. Monitorowanie wydajności

- System powinien monitorować swoją wydajność i obciążenie.
- W przypadku problemów wydajnościowych powinny być generowane alerty.

Priorytet niski

1. Zapewnienie tłumaczenia na inne języki

- System powinien wspierać wielojęzyczność.
- Interfejs i dane powinny być dostępne w różnych językach.

2. Statystyki wyszukiwań

- System powinien zbierać statystyki dotyczące wyszukiwań.
- Statystyki powinny być dostępne dla administratorów.

3. Ocena UI

- Użytkownicy powinni móc oceniać interfejs użytkownika.
- System powinien zbierać opinie na temat użyteczności.

4. Śledzenie aktywności użytkownika w ocenianiu

- System powinien śledzić aktywność użytkowników związaną z ocenianiem zbiorów.
- Informacje te powinny być wykorzystywane do poprawy jakości danych.

5. Ocenianie danych pod kątem przydatności

- Użytkownicy powinni móc oceniać przydatność danych.
- System powinien uwzględniać te oceny w algorytmach wyszukiwania.

6. Zarabianie na reklamach

- System może posiadać mechanizmy monetyzacji poprzez wyświetlanie reklam.
- Reklamy powinny być nieinwazyjne i związane z kontekstem.

7. Wspieranie badań naukowych poprzez dostarczanie zbiorów

- System powinien oferować specjalne funkcje dla użytkowników akademickich.
- Dane powinny być przygotowane w formacie odpowiednim dla celów badawczych.

8. Cachowanie odpowiedzi

- System powinien implementować mechanizmy cachowania dla częstych zapytań.
- Cachowanie powinno optymalizować wydajność i zmniejszać obciążenie bazy danych.

3.4 Cechy jakościowe

Cechy jakościowe systemu

System Zarządzania Otwartymi Danymi Publicznymi powinien spełniać następujące wymagania jakościowe, pogrupowane według priorytetu.

Priorytet wysoki

1. Efektywne zarządzanie zbiorami danych

- System powinien umożliwiać efektywne i ergonomiczne zarządzanie zbiorami danych.
- Interfejs zarządzania powinien być intuicyjny i minimalizować liczbę kroków potrzebnych do wykonania operacji.

2. Zapewnienie stałego dostępu do danych

- System musi zapewniać nieprzerwany dostęp do danych z minimalną liczbą niedostępności.
- Architektura powinna być redundantna, aby zapobiegać przestojom.

3. Time to recovery

- W przypadku awarii, czas przywracania funkcjonalności systemu nie powinien przekraczać 2 godzin.
- Proces odzyskiwania powinien być zautomatyzowany i regularnie testowany.

4. Ochrona przed atakami

- System powinien posiadać zabezpieczenia przed typowymi atakami sieciowymi (DDoS, SQL Injection, XSS).
- Regularne testy penetracyjne i audyty bezpieczeństwa powinny być przeprowadzane.

5. Ciągłość działania

- System powinien działać bez przerw, z minimalnymi planowanymi przerwami technicznymi.
- Aktualizacje systemu powinny odbywać się bez całkowitego wyłączenia usługi.

6. Przezroczystość interfejsu

- Interfejs użytkownika powinien być intuicyjny i łatwy w obsłudze dla użytkowników o różnym poziomie zaawansowania.
- Elementy interfejsu powinny być logicznie pogrupowane i opisane.

7. Sprawne dodawanie nowych źródeł danych

- Proces dodawania nowych źródeł danych powinien być prosty i szybki w implementacji.
- Dodawanie nowego źródła powinno wymagać minimalnej konfiguracji.

8. Zmniejszenie kosztów działania

- System powinien optymalizować wykorzystanie zasobów, aby minimalizować koszty operacyjne.
- Automatyzacja rutynowych operacji powinna redukować koszty utrzymania.

Priorytet średni

1. SLA (poziom dostępności)

- System powinien zapewniać poziom dostępności (SLA) na poziomie 99.99999% (tzw. "seven nines").
- Monitoring dostępności powinien umożliwiać szybką reakcję na problemy.

2. Zminimalizowanie czasu wyszukiwania

- Czas potrzebny na wyszukanie danych powinien być jak najkrótszy.
- Maksymalny czas przeszukiwania zbiorów danych nie powinien przekraczać 20 sekund, nawet przy złożonych zapytaniach.

3. Skalowalność systemu

- Architektura powinna umożliwiać skalowanie poziome i pionowe w zależności od rosnącego obciążenia.
- System powinien automatycznie dostosowywać wykorzystanie zasobów do bieżącego zapotrzebowania.

4. Minimalizacja opóźnień

- System powinien być zoptymalizowany pod kątem minimalizacji opóźnień w dostępie do danych.
- Opóźnienia w przesyłaniu i przetwarzaniu danych powinny być monitorowane i optymalizowane.

5. Wsparcie dla różnych formatów

- System powinien obsługiwać szeroki zakres formatów danych wejściowych i wyjściowych.
- Konwersja między formatami powinna odbywać się bez utraty informacji.

6. Adaptacyjność UI

- Interfejs powinien być w pełni responsywny i dostosowywać się do różnych rozmiarów ekranów (desktop, tablet, telefon).

- Układ elementów interfejsu powinien automatycznie reagować na zmianę rozdzielczości ekranu.

7. Efektywność wprowadzania danych

- Proces wprowadzania nowych danych nie powinien zajmować więcej niż 2 minuty.
- Interfejs wprowadzania danych powinien oferować podpowiedzi i automatyczne uzupełnianie.

8. Archiwizacja danych

- Dane historyczne powinny być archiwizowane w sposób efektywny, nie obciążający głównej bazy danych.
- System archiwizacji powinien umożliwiać szybkie przywracanie archiwalnych danych w razie potrzeby.

9. Szybkość powiadomień

- Czas od wystania do otrzymania powiadomienia nie powinien przekraczać 5 minut.
- System powinien optymalizować dostarczanie powiadomień w zależności od ich priorytetu.

10. Szybkość aktualizacji danych

- Wybrane, krytyczne zbiory danych powinny być aktualizowane w czasie rzeczywistym, z opóźnieniem nieprzekraczającym jednej sekundy.
- Mechanizm aktualizacji powinien być wydajny i nie obciążać nadmiernie systemu.

Priorytet niski

1. Śledzenie aktywności użytkowników

- System powinien śledzić aktywność użytkowników związaną z ocenianiem zbiorów danych.
- Dane o aktywności powinny być analizowane w celu poprawy jakości systemu.

2. Szybkość generowania raportów

- Generowanie standardowych raportów powinno zajmować mniej niż jedną minutę.
- System powinien optymalizować proces generowania raportów poprzez wstępne przetwarzanie danych.

3. Dostępność dla osób z niepełnosprawnościami

- Interfejs powinien spełniać standardy dostępności WCAG 2.1 AA.
- System powinien być kompatybilny z technologiami wspomagającymi używanymi przez osoby z niepełnosprawnościami.

3.5 Słownik

Ograniczenia

Priorytet wysoki

Zapewnienie danych zgodnie ze standardami dostępności

- Wszystkie dane muszą być dostarczane w sposób zgodny ze standardami dostępności (np. WCAG 2.1 AA).

Prezentacja wyników w różnych formatach

- System musi umożliwiać eksport danych w co najmniej trzech formatach: CSV, JSON i XML.

Integracja z CKAN

- System musi być kompatybilny z platformą CKAN i umożliwiać integrację poprzez API.

Kompatybilność systemu

- Minimalne wymagania dla urządzeń mobilnych: Android 10 i iOS 12.

Wsparcie dla Internet Explorer 11

- System musi zapewniać podstawową funkcjonalność w przeglądarce Internet Explorer 11.

Infrastruktura na terenie Unii Europejskiej

- Wszystkie serwery i centra danych muszą znajdować się na terytorium UE.

Zgodność z RODO i NIS2

- System musi spełniać wymagania **RODO** (Rozporządzenie o Ochronie Danych Osobowych) oraz **NIS2** (Dyrektywa w sprawie bezpieczeństwa sieci i informacji).

Priorytet średni

Ograniczenie dostępu do panelu administracyjnego

- Panel administracyjny musi być dostępny wyłącznie z wewnętrznej sieci (intranet).

Priorytet niski

Dostępność dla osób z niepełnosprawnościami

- Interfejs systemu musi być zgodny ze standardami WCAG 2.1 AA i obsługiwać technologie wspomagające.

3.6 Słownik

Słownik

Administracja publiczna: Instytucje rządowe i samorządowe tworzące i udostępniające dane publiczne.

Aktualizacja danych: Proces regularnego sprawdzania dostępności nowych danych, ich pobierania, walidacji i zapisywania w systemie, aby utrzymać bazę danych w stanie aktualności i zgodności z

rzeczywistością. System powinien umożliwiać ciągłą aktualizację danych i automatyczne udostępnianie zaktualizowanych danych.

Analiza danych: Proces przetwarzania i badania danych w celu identyfikacji trendów, wzorców oraz anomalii.

Analiza ruchu: Badanie danych dotyczących aktywności użytkowników w systemie.

API (Interfejs Programowania Aplikacji): Interfejs umożliwiający komunikację i wymianę danych między różnymi systemami lub aplikacjami. System ma udostępniać dane poprzez API.

Archiwizacja danych: Proces przechowywania danych historycznych w celu ich zabezpieczenia i możliwości odtworzenia.

Autentyczność danych: Zapewnienie, że dane są wiarygodne i pochodzą z zaufanego źródła. System powinien zapewniać autentyczność danych.

Autoryzacja: Proces weryfikacji, czy użytkownik ma uprawnienia do wykonania określonej akcji lub dostępu do określonych zasobów.

Automatyzacja: Wykorzystanie systemów do wykonywania procesów biznesowych w celu minimalizacji błędów ludzkich i zwiększenia efektywności.

Awarie: Nieprawidłowe działanie systemu. System powinien monitorować swój stan w celu wykrywania awarii i informować o nich administratora.

Backup danych: Kopia zapasowa danych, tworzona w celu możliwości przywrócenia systemu do poprzedniego stanu w przypadku awarii lub błędów.

Bezpieczeństwo danych: Ochrona danych przed nieautoryzowanym dostępem, utratą lub uszkodzeniem. System ma zapewniać bezpieczeństwo danych i chronić przed atakami.

Błędy danych: Niepoprawne, niekompletne lub błędne informacje znajdujące się w systemie. Użytkownicy powinni mieć możliwość zgłaszania błędów danych.

CKAN: Popularna platforma do zarządzania i udostępniania otwartych danych. System ma być zintegrowany z CKAN i zgodny ze standardami CKAN.

Dane otwarte: Dane publiczne udostępniane w sposób otwarty, zgodnie z określonymi standardami, umożliwiające ich swobodne wykorzystywanie i ponowne użycie.

Dane publiczne: Dane tworzone i udostępniane przez administrację publiczną. Przykłady to rozkłady jazdy, dane o położeniu pojazdów, stężenia zanieczyszczeń, położenia toalet miejskich, godziny dostępności urzędów.

DCAT: Standard metadanych opisujący publiczne dane. System powinien walidować dane pod kątem zgodności ze standardem DCAT i być zgodny z nim.

Dostęp do danych: Możliwość korzystania z danych przez uprawnionych użytkowników. System ma ułatwiać dostęp do różnych źródeł danych dla klientów i zapewniać stały oraz ciągły dostęp do danych.

Dystrybucje zbiorów danych: Konkretnie formy udostępniania zbiorów danych (np. pliki w określonym formacie, dostęp przez API). System ma zarządzać dystrybucjami zbiorów danych.

Efektywność: Zdolność systemu do sprawnego i szybkiego działania, np. minimalizowanie czasu potrzebnego do wyszukiwania danych.

Filtrowanie danych: Proces wybierania danych spełniających określone kryteria. Użytkownicy mają mieć możliwość filtrowania zbiorów danych.

Formaty danych: Sposoby zapisu i organizacji danych (np. JSON, XML, CSV, RDF). System ma udostępniać dane w różnych formatach oraz umożliwiać import i eksport danych w różnych formatach.

Generowanie raportów: Automatyczne tworzenie podsumowań i wizualizacji danych. System ma generować raporty o ruchu.

Historia danych: Zapis zmian danych w czasie, umożliwiający śledzenie ich ewolucji. System powinien utrzymywać historię danych i przedstawiać historię zmian w przystępny sposób.

Integracja z CKAN: Połączenie i współpraca systemu z platformą CKAN.

Interfejs użytkownika (UI): Element systemu umożliwiający interakcję użytkownika z systemem. Interfejs powinien być przejrzysty i adaptowalny do rozmiaru ekranu.

Intranet: Prywatna sieć komputerowa, dostępna tylko dla pracowników organizacji. Dostęp do panelu administracyjnego powinien być ograniczony do intranetu.

Jakość danych: Stopień, w jakim dane są dokładne, kompletne, spójne i aktualne. System ma ulepszać jakość danych i monitorować ją.

Logi systemu: Zapis aktywności systemu, przechowywany w celu audytu i śledzenia błędów.

Metadane: Dane opisujące inne dane, zawierające informacje o ich pochodzeniu, formacie, strukturze itp.. System ma zarządzać metadanymi udostępnionych zbiorów danych.

Monitorowanie systemu: Ciągłe sprawdzanie działania systemu w celu wykrywania problemów, anomalii lub zagrożeń.

Oceny użytkowników: Opinie i oceny zbiorów danych wystawiane przez użytkowników. System ma zbierać i analizować oceny użytkowników oraz prezentować oceny danych.

Otwartość danych (piąty poziom): Zgodność z zasadami otwartych danych, w tym m.in. z formatem RDF (Resource Description Framework), umożliwiającym łączenie danych.

Panel administracyjny: Interfejs służący do zarządzania systemem, dostępny zazwyczaj dla administratorów.

Procesy biznesowe: Zestaw powiązanych działań lub zadań, które razem realizują określony cel biznesowy. Źródła opisują różne procesy biznesowe związane z zarządzaniem danymi.

Przeglądanie danych: Możliwość oglądania i zapoznawania się z zawartością zbiorów danych.

Przejrzystość interfejsu: Zrozumiałość i łatwość obsługi interfejsu użytkownika.

RDF (Resource Description Framework): Standard formatu danych używany w semantycznym Webie, umożliwiający powiązanie różnych zbiorów danych. Zgodność z RDF jest wymogiem piątego poziomu otwartości danych.

Rejestracja nowych źródeł danych: Proces dodawania informacji o nowych źródłach danych do systemu. Może odbywać się poprzez API lub formularz.

Retencja danych: Okres przechowywania danych w systemie.

RODO (Ogólne Rozporządzenie o Ochronie Danych): Przepisy prawne dotyczące ochrony danych osobowych. System powinien być zgodny z RODO.

Rola użytkownika: Określona kategoria użytkowników systemu z przypisanymi uprawnieniami dostępu.

Skalowalność systemu: Zdolność systemu do obsługi rosnącej liczby użytkowników i danych bez utraty wydajności.

SLA (Service Level Agreement): Umowa dotycząca poziomu świadczonych usług, np. dostępności systemu. Źródła wspominają o SLA na poziomie 99.99999%.

Sprawdzanie poziomu dostępu: Weryfikacja uprawnień użytkownika do określonych danych lub funkcji systemu.

Standardy danych: Ustalone wytyczne dotyczące formatu, struktury i jakości danych. Dane muszą spełniać określone standardy, aby móc je udostępniać.

Statystyki wyszukiwań: Dane dotyczące zapytań wyszukiwania użytkowników.

System zarządzania otwartymi danymi publicznymi: System informatyczny mający na celu sprawne zarządzanie danymi publicznymi i ich udostępnianie w sposób otwarty.

Szyfrowanie danych: Kodowanie danych w celu ochrony ich poufności. Dotyczy zarówno danych wrażliwych, jak i przesyłu danych.

Śledzenie historii zmian: Rejestrowanie modyfikacji danych w czasie.

Udostępnianie danych: Udostępnianie zbiorów danych użytkownikom w różnych formatach i poprzez różne kanały (np. aplikacja webowa, API).

Uwierzytelnianie: Proces weryfikacji tożsamości użytkownika.

Walidacja danych: Sprawdzanie poprawności danych pod kątem zgodności z określonymi regułami i standardami.

Wersjonowanie danych: Zarządzanie różnymi wersjami zbiorów danych i dystrybucji.

Wyszukiwanie danych: Proces znajdowania danych spełniających określone kryteria. System ma minimalizować czas potrzebny do wyszukiwania danych i prezentować wyniki wyszukiwania w różnych formatach.

Zabezpieczenie przed atakami: Działania mające na celu ochronę systemu przed cyberatakami i nieautoryzowanym dostępem.

Zgodność z RODO i NIS2: Przestrzeganie przepisów dotyczących ochrony danych osobowych (RODO) oraz bezpieczeństwa systemów sieciowych i informacyjnych (NIS2).

Zgłaszanie błędów: Proces umożliwiający użytkownikom informowanie o nieprawidłowościach w danych.

Zbiory danych: Kolekcje danych udostępnianych przez system. System ma umożliwiać wygodne wprowadzanie, przeglądanie, filtrowanie i wyszukiwanie zbiorów danych oraz zarządzanie nimi.

4. Wymagania użytkownika

4.1 Wymagania funkcjonalne

4.1.1 Zarządzanie zbiorami danych

Diagram



Przypadki użycia

UC0001 Wyświetlenie zbiorów danych

nazwa: Wyświetlanie zbiorów danych

aktorzy: Klient/Użytkownik danych, System

opis: System prezentuje listę wszystkich dostępnych zbiorów danych. Prezentacja powinna być przejrzysta i umożliwiać sortowanie. Użytkownik ma możliwość zapoznawania się z zawartością zbiorów danych.

kroki:

1. **Klient/Użytkownik danych** klika przycisk 'Wyświetl dostępne zbiory'.
2. **System** pobiera z bazy danych informacje o wszystkich dostępnych zbiorach danych.
3. **System** generuje przejrzystą listę zbiorów danych.
4. **System** wyświetla listę zbiorów danych.

alt 1 Sortowanie zbiorów danych:

1. **Klient/Użytkownik danych** wybiera opcję sortowania według danego kryterium.
2. **System** sortuje aktualnie wyświetlaną listę zbiorów danych zgodnie z wybranym kryterium.
3. **System** odświeża widok, prezentując posortowaną listę zbiorów danych.

alt 2 awaria bazy danych:

1. **Klient/Użytkownik danych** klika przycisk 'Wyświetl dostępne zbiory'.
2. **System** próbuje połączyć się z bazą danych w celu pobrania informacji o zbiorach danych.
3. **System** wykrywa awarię bazy danych (brak połączenia, błędy w odpowiedzi).
4. **System** wyświetla Klientowi/Użytkownikowi danych komunikat o błędzie informujący o niedostępności danych z powodu awarii bazy danych.

UC0002 Wyszukiwanie zbiorów danych

nazwa: Wyszukiwanie zbiorów danych

aktorzy: Klient/Użytkownik danych, System

opis: System umożliwia użytkownikowi wyszukiwanie konkretnych zbiorów danych na podstawie podanych kryteriów. Wyniki wyszukiwania powinny być czytelne.

kroki:

1. **Klient/Użytkownik danych** wpisuje frazę wyszukiwania lub wybiera kryteria wyszukiwania.
2. **System** przetwarza zapytanie i przeszukuje dostępne zbiory danych.
3. **System** generuje listę wyników zgodną z podanymi kryteriami.
4. **System** wyświetla wyniki wyszukiwania.

alt 1 Brak wyników wyszukiwania:

1. **System** nie znajduje zbiorów danych spełniających podane kryteria.
2. **System** wyświetla komunikat informujący o braku wyników oraz sugeruje zmianę kryteriów wyszukiwania.

UC0003 Filtrowanie zbiorów danych

nazwa: Filtrowanie zbiorów danych

aktorzy: Klient/Użytkownik danych, System

opis: System umożliwia użytkownikowi zawężenie listy dostępnych zbiorów danych poprzez zastosowanie dodatkowych filtrów.

kroki:

1. **Klient/Użytkownik danych** wybiera kryteria filtrowania dostępnych zbiorów danych.
2. **System** przetwarza wybrane kryteria i filtruje dostępne zbiory danych.
3. **System** generuje i wyświetla zaktualizowaną listę zbiorów danych zgodną z zastosowanymi filtrami.

alt 1 Brak wyników po zastosowaniu filtrów:

1. **System** nie znajduje zbiorów danych spełniających zastosowane filtry.
2. **System** wyświetla komunikat o braku wyników i sugeruje zmianę kryteriów filtrowania.

UC0004 Wizualizacja zbioru danych

nazwa: Wizualizacja zbiorów danych

aktorzy: Klient/Użytkownik danych, System

opis: System umożliwia użytkownikowi graficzne przedstawienie danych w wybranej formie (np. wykresy, tabele, diagramy) dla lepszej analizy i interpretacji.

kroki:

1. **Klient/Użytkownik danych** wybiera zbiór danych do wizualizacji.
2. **System** prezentuje dostępne opcje wizualizacji (np. wykresy słupkowe, liniowe, kołowe).
3. **Klient/Użytkownik danych** wybiera preferowany typ wizualizacji.
4. **System** generuje i wyświetla wizualizację wybranego zbioru danych.

alt 1 Brak wystarczających danych do wizualizacji:

1. **System** wykrywa, że wybrany zbiór danych nie zawiera wystarczającej ilości informacji do wygenerowania wizualizacji.
2. **System** wyświetla komunikat informujący o niemożliwości wygenerowania wizualizacji i sugeruje wybór innego zbioru danych lub innej metody prezentacji.

UC0005 Prezentacja historii zmian zbioru danych

nazwa: Prezentacja historii zmian zbioru danych

aktorzy: Klient/Użytkownik danych, System

opis: System umożliwia użytkownikowi przeglądanie historii zmian wybranego zbioru danych, w tym informacji o modyfikacjach, czasie ich dokonania oraz osobach, które je wykonały.

kroki:

1. **Klient/Użytkownik danych** wybiera zbiór danych, dla którego chce zobaczyć historię zmian.
2. **System** pobiera z bazy danych informacje o zmianach dokonanych w wybranym zbiorze.
3. **System** generuje listę zawierającą szczegóły każdej zmiany (np. data, autor, typ zmiany).
4. **System** wyświetla użytkownikowi historię zmian w czytelnej formie.

alt 1 Brak historii zmian dla zbioru danych:

1. **System** wykrywa, że dla wybranego zbioru danych nie ma zarejestrowanych zmian.
2. **System** wyświetla komunikat informujący użytkownika o braku historii zmian.

UC0006 Dodawanie nowych zbiorów danych

nazwa: Dodawanie nowych zbiorów danych

aktorzy: Dostawca danych, System

opis: System umożliwia dostawcy danych dodanie nowego zbioru do systemu, w tym wprowadzenie jego opisu oraz metadanych. Proces dodawania obejmuje walidację danych, która jest realizowana jako osobny przypadek użycia (**include: Walidacja Danych**).

kroki:

1. **Dostawca danych** wybiera opcję dodania nowego zbioru danych.
2. **System** wyświetla formularz do wprowadzenia podstawowych informacji o zbiorze (np. nazwa, opis, kategorie, format pliku).
3. **Dostawca danych** wypełnia formularz i przesyła plik zawierający dane.
4. **System** wykonuje przypadek użycia **Walidacja Danych**.
5. **System** zapisuje nowy zbiór danych w bazie i rejestruje go w katalogu zbiorów.
6. **System** wyświetla komunikat potwierdzający dodanie nowego zbioru danych.

alt 1 Niepowodzenie walidacji danych:

1. **System** wykrywa, że przestany zbiór danych zawiera błędy w ramach przypadku użycia **Walidacja Danych**.
2. **System** wyświetla komunikat o błędach i prosi dostawcę danych o poprawienie i ponowne przestanie zbioru danych.

UC0007 Edycja zbioru danych

nazwa: Edycja zbioru danych

aktorzy: Dostawca danych, System

opis: System umożliwia dostawcy danych edytowanie istniejącego zbioru danych. Proces edycji obejmuje możliwość zmiany opisu zbioru, kategorii, formatu pliku oraz innych metadanych. Zmiany wprowadzane przez dostawcę są walidowane przed zapisaniem w systemie. Walidacja danych jest realizowana jako osobny przypadek użycia (**include: Walidacja Danych**).

kroki:

1. **Dostawca danych** wybiera opcję edytowania istniejącego zbioru danych.
2. **System** wyświetla formularz z obecnymi informacjami o zbiorze (np. nazwa, opis, kategorie, format pliku).
3. **Dostawca danych** wprowadza zmiany w formularzu i przesyła zaktualizowane dane.
4. **System** wykonuje przypadek użycia **Walidacja Danych**.
5. **System** zapisuje zaktualizowany zbiór danych w bazie i rejestruje zmiany w katalogu zbiorów.
6. **System** wyświetla komunikat potwierdzający pomyślną edycję zbioru danych.

alt 1 Niepowodzenie walidacji danych:

1. **System** wykrywa, że wprowadzone dane zawierają błędy (np. niepoprawny format, brak wymaganych informacji) w ramach przypadku użycia **Walidacja Danych**.
2. **System** wyświetla komunikat o błędach i prosi dostawcę danych o poprawienie i ponowne przestanie zaktualizowanego zbioru danych.

UC0008 Walidacja danych

nazwa: Walidacja danych

aktorzy: System

opis: System sprawdza poprawność danych przesyłanych przez dostawcę danych. Walidacja obejmuje kontrolę formatu pliku, poprawności wymaganych pól oraz spójności danych. Celem jest zapewnienie, że przestane dane są zgodne z wymaganiami systemu przed ich zapisaniem lub przetworzeniem.

kroki:

1. **System** otrzymuje dane od dostawcy danych.
2. **System** sprawdza, czy dane są w poprawnym formacie.
3. **System** weryfikuje, czy wszystkie wymagane pola są obecne i mają poprawny typ danych.
4. **System** sprawdza spójność danych (np. czy daty są w odpowiednim formacie, czy wartości mieszczą się w dozwolonym zakresie).
5. **System** zwraca wynik walidacji:
 - Jeśli dane są poprawne, proces przechodzi do zapisu lub dalszego przetwarzania.
 - Jeśli dane zawierają błędy, system zgłasza je dostawcy danych.

alt 1 Niepoprawne dane:

1. **System** wykrywa, że dane zawierają błędy (np. niepoprawny format, brak wymaganych pól, błędne wartości).
2. **System** wyświetla komunikat o błędach i prosi dostawcę danych o poprawienie błędów i ponowne przestanie danych.

UC0009 Importowanie danych

nazwa: Importowanie danych

aktorzy: Dostawca danych, System

opis: System umożliwia dostawcy danych importowanie zbiorów danych do systemu. Importowanie danych obejmuje przesyłanie plików lub danych w odpowiednim formacie, które zostaną zapisane w systemie. Proces ten może obejmować walidację danych, kontrolę jakości oraz automatyczne mapowanie danych do struktury systemu.

kroki:

1. **Dostawca danych** wybiera opcję importu danych w systemie.
2. **System** wyświetla formularz lub interfejs do przestania pliku z danymi.
3. **Dostawca danych** wybiera plik z danymi do importu i przesyła go do systemu.

4. **System** wykonuje przypadek użycia **Walidacja danych** w celu sprawdzenia poprawności przestanych danych.
5. **System** mapuje dane z przestanego pliku do odpowiednich struktur danych w systemie.
6. **System** zapisuje zaimportowane dane w bazie danych.
7. **System** wyświetla komunikat potwierdzający pomyślne zaimportowanie danych.

alt 1 Niepowodzenie walidacji danych:

1. **System** wykrywa, że dane w przestanym pliku zawierają błędy (np. niepoprawny format, brak wymaganych pól).
2. **System** wyświetla komunikat o błędach i prosi dostawcę danych o poprawienie pliku oraz ponowne przesłanie danych.

alt 2 Niepoprawny format pliku:

1. **Dostawca danych** przesyła plik w nieobsługiwanym formacie (np. plik tekstowy, niespójny format CSV).
2. **System** wykrywa niepoprawny format pliku.
3. **System** wyświetla komunikat informujący o błędzie formatu pliku i prosi o przesłanie pliku w obsługiwanym formacie.

UC0010 Przypisywanie schematów do zbiorów danych

nazwa: Przypisywanie schematów do zbiorów danych

aktorzy: Dostawca danych, System

opis: System umożliwia dostawcy danych przypisanie schematu do zbioru danych. Schemat określa strukturę danych, w tym wymagane pola, formaty oraz inne zasady, które muszą być przestrzegane przy wprowadzaniu danych. Przypisanie schematu jest kluczowe, aby zapewnić spójność danych i umożliwić ich prawidłowe przetwarzanie w systemie.

kroki:

1. **Dostawca danych** wybiera opcję przypisania schematu do zbioru danych.
2. **System** wyświetla listę dostępnych schematów, które mogą być przypisane do zbioru danych.
3. **Dostawca danych** wybiera odpowiedni schemat z listy.
4. **System** przypisuje wybrany schemat do zbioru danych.
5. **System** zapisuje przypisanie schematu w bazie danych, aktualizując metadane zbioru.
6. **System** wyświetla komunikat potwierdzający przypisanie schematu do zbioru danych.

alt 1 Brak dostępnych schematów:

1. **Dostawca danych** wybiera opcję przypisania schematu, ale system nie ma dostępnych schematów w bazie.
2. **System** wyświetla komunikat informujący o braku dostępnych schematów do przypisania.

alt 2 Błąd przy przypisywaniu schematu:

1. **Dostawca danych** wybiera schemat i system próbuje go przypisać do zbioru danych.
2. **System** wykrywa błąd (np. konflikt schematu z istniejącymi danymi).

3. **System** wyświetla komunikat o błędzie i informuje dostawcę danych o konieczności wyboru innego schematu.

UC0011 Eksportowanie danych w różnych formatach

nazwa: Eksportowanie danych w różnych formatach

aktorzy: Klient/Użytkownik danych, System

opis: System umożliwia użytkownikowi eksportowanie danych w różnych formatach (np. CSV, JSON, XML, Excel). Proces eksportu pozwala na wygodne pobranie danych z systemu i ich dalsze przetwarzanie lub analiza w wybranym formacie. Użytkownik może wybrać format, który najlepiej odpowiada jego potrzebom.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję eksportu danych.
2. **System** wyświetla dostępne formaty eksportu (np. CSV, JSON, XML, Excel).
3. **Klient/Użytkownik danych** wybiera preferowany format eksportu.
4. **System** generuje plik w wybranym formacie, zawierający dane do pobrania.
5. **System** umożliwia użytkownikowi pobranie wygenerowanego pliku.
6. **System** wyświetla komunikat potwierdzający pomyślne zakończenie eksportu.

alt 1 Brak danych do eksportu:

1. **Klient/Użytkownik danych** wybiera opcję eksportu danych.
2. **System** wykrywa, że brak jest dostępnych danych do eksportu (np. brak wyników dla wybranych filtrów).
3. **System** wyświetla komunikat informujący o braku danych do eksportu.

alt 2 Błąd podczas generowania pliku:

1. **Klient/Użytkownik danych** wybiera format eksportu.
2. **System** napotyka błąd podczas generowania pliku (np. problemy z formatowaniem, brak uprawnień).
3. **System** wyświetla komunikat o błędzie i informuje użytkownika o problemach z eksportem.

UC0012 Usuwanie zbioru danych

nazwa: Usuwanie zbiorów danych

aktorzy: Administrator systemu, System

opis: System umożliwia Administratorowi systemu usunięcie zbiorów danych z bazy. Usunięcie zbioru danych jest ostateczną akcją, która trwale usuwa dane z systemu. Proces usuwania może wymagać potwierdzenia, aby zapobiec przypadkowemu usunięciu danych.

kroki:

1. **Administrator systemu** wybiera opcję usunięcia zbioru danych.
2. **System** wyświetla listę dostępnych zbiorów danych do usunięcia.
3. **Administrator systemu** wybiera zbiór danych, który chce usunąć.
4. **System** wyświetla komunikat potwierdzający, informując, że operacja usunięcia jest nieodwracalna.

5. **Administrator systemu** potwierdza chęć usunięcia zbioru danych.
6. **System** usuwa wskazany zbiór danych z bazy.
7. **System** wyświetla komunikat potwierdzający pomyślne usunięcie zbioru danych.

alt 1 Brak uprawnień:

1. **Administrator systemu** próbuje usunąć zbiór danych.
2. **System** sprawdza uprawnienia administratora.
3. **System** wykrywa brak wymaganych uprawnień (np. brak uprawnień do usunięcia danych).
4. **System** wyświetla komunikat informujący o braku uprawnień do wykonania operacji.

alt 2 Błąd podczas usuwania:

1. **Administrator systemu** wybiera zbiór danych do usunięcia.
2. **System** napotyka błąd podczas próby usunięcia danych (np. problemy z bazą danych, brak dostępu do danych).
3. **System** wyświetla komunikat o błędzie i informuje administratora o problemach z usuwaniem zbioru danych.

UC0013 Archiwizacja zbiorów danych

nazwa: Archiwizacja zbiorów danych

aktorzy: Administrator systemu, System

opis: System umożliwia Administratorowi systemu archiwizowanie zbiorów danych, które nie są już aktywnie używane, ale muszą zostać zachowane w celu przechowywania lub przyszłego wykorzystania. Archiwizacja polega na przeniesieniu danych do specjalnej przestrzeni archiwalnej, gdzie będą mogły być bezpiecznie przechowywane. Po archiwizacji, dane są niedostępne do regularnego użytku, ale mogą być odzyskane, jeśli zajdzie taka potrzeba.

kroki:

1. **Administrator systemu** wybiera opcję archiwizacji zbioru danych.
2. **System** wyświetla listę dostępnych zbiorów danych, które mogą być archiwizowane.
3. **Administrator systemu** wybiera zbiór danych, który ma zostać zarchiwizowany.
4. **System** wyświetla komunikat potwierdzający, że zbiór danych zostanie przeniesiony do archiwum.
5. **Administrator systemu** potwierdza chęć archiwizacji zbioru danych.
6. **System** przenosi zbiór danych do przestrzeni archiwalnej, zmieniając jego status na "zarchiwizowany".
7. **System** wyświetla komunikat potwierdzający pomyślną archiwizację zbioru danych.

alt 1 Brak uprawnień:

1. **Administrator systemu** próbuje archiwizować zbiór danych.
2. **System** sprawdza uprawnienia administratora.
3. **System** wykrywa brak wymaganych uprawnień (np. brak uprawnień do archiwizacji).
4. **System** wyświetla komunikat informujący o braku uprawnień do wykonania operacji.

alt 2 Błąd podczas archiwizacji:

1. **Administrator systemu** wybiera zbiór danych do archiwizacji.
2. **System** napotyka błąd podczas próby archiwizacji (np. problemy z przestrzenią archiwalną lub dostępem do danych).
3. **System** wyświetla komunikat o błędzie i informuje administratora o problemach z archiwizowaniem zbioru danych.

UC0014 Przywracanie zbiorów danych

nazwa: Przywracanie zbiorów danych

aktorzy: Administrator systemu, System

opis: System umożliwia Administratorowi systemu przywracanie zbiorów danych, które zostały wcześniej zarchiwizowane. Przywracanie danych polega na przeniesieniu zarchiwizowanych zbiorów danych z przestrzeni archiwalnej do aktywnej bazy danych, aby mogły być ponownie używane w systemie. Proces przywracania danych może obejmować weryfikację integralności danych przed ich ponownym udostępnieniem.

kroki:

1. **Administrator systemu** wybiera opcję przywrócenia zbioru danych.
2. **System** wyświetla listę dostępnych zarchiwizowanych zbiorów danych, które mogą zostać przywrócone.
3. **Administrator systemu** wybiera zbiór danych, który ma zostać przywrócony.
4. **System** weryfikuje integralność danych przed przywróceniem (np. sprawdza, czy dane nie zostały uszkodzone w procesie archiwizacji).
5. **Administrator systemu** potwierdza chęć przywrócenia zbioru danych.
6. **System** przywraca zbiór danych z przestrzeni archiwalnej do aktywnej bazy danych.
7. **System** zmienia status zbioru danych na "aktywny" i umożliwia jego dalsze wykorzystanie.
8. **System** wyświetla komunikat potwierdzający pomyślne przywrócenie zbioru danych.

alt 1 Brak uprawnień:

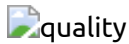
1. **Administrator systemu** próbuje przywrócić zbiór danych.
2. **System** sprawdza uprawnienia administratora.
3. **System** wykrywa brak wymaganych uprawnień (np. brak uprawnień do przywracania danych).
4. **System** wyświetla komunikat informujący o braku uprawnień do wykonania operacji.

alt 2 Błąd podczas przywracania:

1. **Administrator systemu** wybiera zbiór danych do przywrócenia.
2. **System** napotyka błąd podczas próby przywrócenia danych (np. brak dostępu do archiwum, problemy z integracją danych).
3. **System** wyświetla komunikat o błędzie i informuje administratora o problemach z przywracaniem zbioru danych.

4.1.2 Zarządzanie jakością danych

Diagram



Przypadki użycia

UC2001 Zgłoszenie błędów danych

nazwa: Zgłoszenie błędów danych

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia użytkownikowi zgłoszenie błędu w danych. Zgłoszenie jest rejestrowane w celu późniejszej weryfikacji przez administratora lub dedykowany zespół ds. jakości danych.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Zgłoś błąd danych".
 2. **System** wyświetla formularz zgłoszenia błędu.
 3. **Klient/Użytkownik danych** wypełnia formularz, podając opis błędu oraz ewentualne szczegóły.
 4. **Klient/Użytkownik danych** zatwierdza przestanie zgłoszenia.
 5. **System** rejestruje zgłoszenie błędu i wyświetla komunikat potwierdzający.
-

UC2002 Zatwierdzenie zgłoszonych błędów danych

nazwa: Zatwierdzenie zgłoszonych błędów danych

aktorzy: Administrator systemu, System

opis:

Administrator weryfikuje zgłoszone błędy danych i zatwierdza je, jeśli zgłoszenie jest uzasadnione, co umożliwia dalsze działania naprawcze.

kroki:

1. **Administrator systemu** loguje się do systemu i przechodzi do sekcji zgłoszeń błędów.
 2. **System** prezentuje listę oczekujących zgłoszeń.
 3. **Administrator systemu** wybiera zgłoszenie do weryfikacji.
 4. **System** wyświetla szczegóły zgłoszenia.
 5. **Administrator systemu** zatwierdza zgłoszenie.
 6. **System** rejestruje zatwierdzenie i aktualizuje status zgłoszenia.
-

UC2003 Odrzucenie zgłoszenia i powiadomienie użytkownika

nazwa: Odrzucenie zgłoszenia błędu i powiadomienie użytkownika

aktorzy: Administrator systemu, System

opis:

W przypadku uznania zgłoszenia za nieuzasadnione, administrator odrzuca zgłoszenie i powiadamia

użytkownika o przyczynach odrzucenia.

kroki:

1. **Administrator systemu** przegląda zgłoszenie błędu.
2. **System** prezentuje szczegóły zgłoszenia.
3. **Administrator systemu** wybiera opcję "Odrzuć zgłoszenie".
4. **System** rejestruje decyzję o odrzuceniu.
5. **System** generuje powiadomienie z uzasadnieniem decyzji.
6. **System** wysyła powiadomienie do Klienta/Użytkownika danych.

alt 1 – Problem z powiadomieniem:

1. **System** wykrywa problem z wysyłką powiadomienia (np. niepoprawny adres email).
 2. **System** loguje błąd i informuje administratora o problemie.
-

UC2004 Wykrywanie anomalii

nazwa: Wykrywanie anomalii w danych

aktorzy: System

opis:

System automatycznie analizuje dane, identyfikując nieprawidłowości lub odstępstwa od normy przy użyciu algorytmów wykrywania anomalii.

kroki:

1. **System** okresowo analizuje dane w zbiorach.
2. **System** stosuje algorytmy wykrywania anomalii.
3. **System** rejestruje zdarzenie w przypadku wykrycia nieprawidłowości.
4. **System** generuje alert o wykrytej anomalii.

alt 1 – Brak wykrycia anomalii:

1. **System** nie generuje alertu, gdy dane mieszczą się w ustalonych normach.

alt 2 – Błąd algorytmu wykrywania:

1. **System** loguje błąd w algorytmie i informuje administratora o konieczności interwencji.
-

UC2005 Wyświetlanie wykrytych anomalii

nazwa: Wyświetlanie wykrytych anomalii

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia użytkownikowi przeglądanie listy wykrytych anomalii, prezentując szczegóły takie jak rodzaj anomalii, data wykrycia oraz miejsce wystąpienia.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Wyświetl anomalie".
2. **System** pobiera listę wykrytych anomalii z bazy danych.
3. **System** generuje czytelną listę z kluczowymi informacjami.
4. **System** wyświetla listę wykrytych anomalii.

alt 1 – Brak wykrytych anomalii:

1. **System** wyświetla komunikat informujący o braku wykrytych anomalii.
-

UC2006 Walidacja istnienia zbiorów danych

nazwa: Walidacja istnienia zbiorów danych

aktorzy: System

opis:

System sprawdza, czy wymagane zbiory danych są dostępne przed rozpoczęciem operacji przetwarzania lub analizy.

kroki:

1. **System** otrzymuje zapytanie dotyczące operacji na danych.
2. **System** przeszukuje bazę danych w celu potwierdzenia istnienia wymaganych zbiorów.
3. **System** kontynuuje operację, jeśli wszystkie zbiory są dostępne.
4. **System** przerywa operację i loguje błąd, jeśli brak jest jednego z wymaganych zbiorów.

alt 1 – Brak wymaganego zbioru:

1. **System** wyświetla komunikat błędu informujący o braku niezbędnego zbioru danych.
-

4.1.3 Zarządzanie użytkownikami i dostępem**Diagram****Przypadki użycia****UC0021 Logowanie do systemu**

nazwa: Logowanie do systemu

aktorzy: Użytkownik systemu, System **opis:** System umożliwia użytkownikowi zalogowanie się do aplikacji za pomocą poprawnych danych uwierzytelniających (login i hasło). Proces logowania obejmuje weryfikację danych użytkownika oraz przyznanie dostępu do systemu w przypadku poprawnej autoryzacji.

kroki:

1. **Użytkownik systemu** wprowadza login i hasło w odpowiednich polach formularza.
2. **Użytkownik systemu** zatwierdza logowanie.
3. **System** weryfikuje poprawność wprowadzonych danych uwierzytelniających.
4. **System** sprawdza, czy konto użytkownika jest aktywne.
5. **System** przyznaje dostęp do systemu i przekierowuje użytkownika na stronę główną.

alt 1 Niepoprawne dane uwierzytelniające:

1. **System** wykrywa, że wprowadzone dane uwierzytelniające są niepoprawne.
2. **System** wyświetla komunikat o błędzie i prosi użytkownika o ponowne wprowadzenie danych.

alt 2 Konto zablokowane:

1. **System** wykrywa, że konto użytkownika zostało zablokowane (np. z powodu zbyt wielu nieudanych prób logowania).
2. **System** wyświetla komunikat informujący o blokadzie konta i sugeruje kontakt z administratorem.

UC0022 Zmiana hasła użytkownika

nazwa: Zmiana hasła użytkownika

aktorzy: Użytkownik systemu, System

opis:

System umożliwia użytkownikowi zmianę hasła na nowe. Proces obejmuje weryfikację starego hasła, wprowadzenie nowego hasła oraz jego zapisanie w systemie.

kroki:

2. **Użytkownik systemu** przechodzi do sekcji zmiany hasła.
3. **Użytkownik systemu** wprowadza aktualne hasło w odpowiednim polu.
4. **Użytkownik systemu** wprowadza nowe hasło. 4. **Użytkownik systemu** zatwierdza zmianę hasła. 5. **System** weryfikuje poprawność aktualnego hasła.
6. **System** sprawdza zgodność nowego hasła z zasadami bezpieczeństwa. 7. **System** zapisuje nowe hasło i wyświetla komunikat o pomyślnej zmianie hasła.

alt 1 Niepoprawne aktualne hasło:

1. **System** wykrywa, że wprowadzone aktualne hasło jest niepoprawne.
2. **System** wyświetla komunikat o błędzie i prosi użytkownika o ponowne wprowadzenie aktualnego hasła.

alt 2 Nowe hasło nie spełnia wymagań bezpieczeństwa:

1. **System** wykrywa, że nowe hasło nie spełnia wymagań bezpieczeństwa.
2. **System** wyświetla komunikat z prośbą o wprowadzenie nowego hasła zgodnego z zasadami bezpieczeństwa.

UC0023 Dodanie użytkownika

nazwa: Dodanie użytkownika

aktorzy: Administrator systemu, System

opis:

System umożliwia administratorowi dodanie nowego użytkownika do systemu. Proces obejmuje wprowadzenie danych użytkownika, przypisanie odpowiednich ról oraz zapisanie informacji w systemie.

kroki: 3. **Administrator systemu** wybiera opcję dodania nowego użytkownika.

4. **Administrator systemu** wprowadza dane nowego użytkownika.

5. **Administrator systemu** przypisuje role i uprawnienia nowemu użytkownikowi.

5. **Administrator systemu** zatwierdza dodanie użytkownika.

6. **System** weryfikuje poprawność wprowadzonych danych.

7. **System** zapisuje dane nowego użytkownika w bazie danych.

8. **System** wyświetla komunikat o pomyślnym dodaniu użytkownika.

alt 1 Niepoprawne dane użytkownika:

1. **System** wykrywa, że wprowadzone dane użytkownika są niepoprawne.

2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie danych.

alt 2 Konflikt loginu:

1. **System** wykrywa, że wprowadzony login jest już zajęty przez innego użytkownika.

2. **System** wyświetla komunikat o konflikcie loginu.

UC0024 Przyznanie dostępu do systemu

nazwa: Przyznanie dostępu do systemu

aktorzy: Administrator systemu, System

opis:

System umożliwia administratorowi przyznanie dostępu do systemu wybranemu użytkownikowi. Proces obejmuje wybór użytkownika, określenie poziomu dostępu oraz zapisanie zmian w systemie.

kroki:

2. **Administrator systemu** wybiera opcję przyznania dostępu użytkownikom.

3. **Administrator systemu** wybiera użytkownika, któremu chce przyznać dostęp.

4. **Administrator systemu** określa poziom dostępu. 5. **Administrator systemu** zatwierdza zmiany.

6. **System** weryfikuje poprawność wprowadzonych danych.

7. **System** zapisuje zmiany w bazie danych.

8. **System** wyświetla komunikat o pomyślnym przyznaniu dostępu.

alt 1 Niepoprawne dane:

1. **System** wykrywa, że wprowadzone dane są niepoprawne.

2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie danych.

alt 2 Użytkownik już posiada dostęp:

1. **System** wykrywa, że użytkownik już posiada przyznany dostęp na wybranym poziomie.

2. **System** wyświetla komunikat informujący o istniejącym dostępie.

UC0025 Usunięcie użytkownika

nazwa: Usunięcie użytkownika

aktorzy: Administrator systemu, System

opis:

System umożliwia administratorowi usunięcie użytkownika z systemu. Proces obejmuje wprowadzenie danych użytkownika i usunięcie go z systemu.

kroki: 3. **Administrator systemu** wybiera opcję usunięcia użytkownika.

4. **Administrator systemu** wprowadza dane użytkownika.

5. **Administrator systemu** zatwierdza usunięcie użytkownika.

6. **System** weryfikuje poprawność wprowadzonych danych.

7. **System** zapisuje usuwa dane użytkownika z bazy danych.

8. **System** wyświetla komunikat o pomyślnym usnięciu użytkownika.

alt 1 Niepoprawne dane użytkownika:

1. **System** wykrywa, że wprowadzone dane użytkownika są niepoprawne.

2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie danych.

UC0026 Usunięcie dostępu do systemu

nazwa: Usunięcie dostępu do systemu

aktorzy: Administrator systemu, System

opis:

System umożliwia administratorowi usunięcie dostępu do systemu wybranemu użytkownikowi. Proces obejmuje wybór użytkownika, określenie poziomu dostępu oraz zapisanie zmian w systemie.

kroki:

2. **Administrator systemu** wybiera opcję usunięcia dostępu użytkownikom.

3. **Administrator systemu** wybiera użytkownika, któremu chce usunąć dostęp.

4. **Administrator systemu** określa poziom dostępu. 5. **Administrator systemu** zatwierdza zmiany.

6. **System** weryfikuje poprawność wprowadzonych danych.

7. **System** zapisuje zmiany w bazie danych. 8. **System** wyświetla komunikat o pomyślnym przyznaniu dostępu.

alt 1 Niepoprawne dane:

1. **System** wykrywa, że wprowadzone dane są niepoprawne.

2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie danych.

alt 2 Użytkownik nie posiada usuwanego dostępu:

1. **System** wykrywa, że użytkownik nie posiada dostępu na wybranym poziomie.

2. **System** wyświetla komunikat informujący o nieistniejącym dostępie.

UC0027 Audyt dostępu

nazwa: Audyt dostępu

aktorzy: Administrator systemu, System

opis:

System umożliwia administratorowi przeprowadzenie audytu dostępu użytkowników do systemu. Proces obejmuje generowanie raportu zawierającego informacje o poziomach dostępu użytkowników oraz ich aktywności.

kroki:

- 2. **Administrator systemu** przechodzi do sekcji audytu dostępu.
- 3. **Administrator systemu** wybiera zakres danych do audytu.
- 4. **Administrator systemu** zatwierdza żądanie wygenerowania raportu.
- 5. **System** przetwarza żądanie i generuje raport audytu dostępu.
- 6. **System** wyświetla raport audytu dostępu administratorowi.

alt 1 Brak danych do audytu:

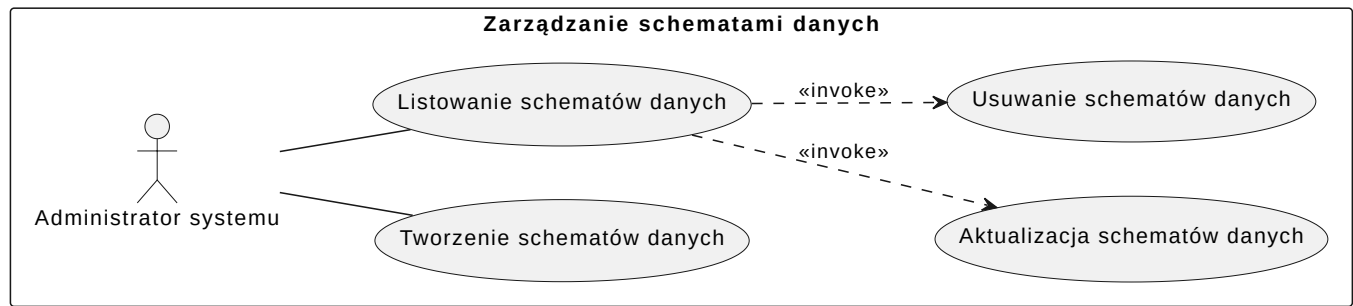
- 1. **System** wykrywa brak danych spełniających kryteria audytu.
- 2. **System** wyświetla komunikat informujący o braku danych do audytu.

alt 2 Błąd podczas generowania raportu:

- 1. **System** napotyka błąd podczas generowania raportu.
- 2. **System** wyświetla komunikat o błędzie.

4.1.4 Zarządzanie schematami danych

Diagram



Przypadki użycia

UC0028 Tworzenie schematów danych

nazwa: Tworzenie schematów danych

aktorzy: Administrator systemu, System, CKAN

opis:

System umożliwia administratorowi tworzenie nowych schematów danych. Proces obejmuje wprowadzenie szczegółów schematu, zapisanie go w systemie oraz synchronizację z platformą CKAN.

kroki:

- 3. **Administrator systemu** wybiera opcję tworzenia nowego schematu danych.
- 4. **Administrator systemu** wprowadza szczegóły schematu.

5. **Administrator systemu** zatwierdza utworzenie schematu.
6. **System** weryfikuje poprawność wprowadzonych danych.
7. **System** zapisuje nowy schemat danych w bazie danych.
8. **System** synchronizuje schemat z platformą CKAN, przesyłając odpowiednie dane.
9. **System** wyświetla komunikat o pomyślnym utworzeniu schematu danych.

alt 1 Niepoprawne dane schematu:

1. **System** wykrywa, że wprowadzone dane schematu są niepoprawne.
2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie danych.

alt 2 Błąd synchronizacji z CKAN:

1. **System** napotyka błąd podczas synchronizacji z platformą CKAN.
2. **System** zapisuje schemat lokalnie i wyświetla komunikat o błędzie synchronizacji.

UC0029 Aktualizacja schematów danych

nazwa: Aktualizacja schematów danych

aktorzy: Administrator systemu, System, CKAN

opis:

System umożliwia administratorowi aktualizację istniejących schematów danych. Proces obejmuje edycję szczegółów schematu, zapisanie zmian w systemie oraz synchronizację z platformą CKAN.

kroki:

3. **Administrator systemu** wybiera schemat danych do edycji.
4. **Administrator systemu** wprowadza zmiany w szczegółach schematu. 5. **Administrator systemu** zatwierdza zmiany.
6. **System** weryfikuje poprawność wprowadzonych danych.
7. **System** zapisuje zmiany w bazie danych.
8. **System** synchronizuje zmiany z platformą CKAN.
9. **System** wyświetla komunikat o pomyślnej aktualizacji schematu danych.

alt 1 Niepoprawne dane schematu:

1. **System** wykrywa, że wprowadzone dane schematu są niepoprawne.
2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie danych.

alt 2 Błąd synchronizacji z CKAN:

1. **System** napotyka błąd podczas synchronizacji z platformą CKAN.
2. **System** zapisuje zmiany lokalnie i wyświetla komunikat o błędzie synchronizacji.

UC0030 Listowanie schematów danych

nazwa: Listowanie schematów danych

aktorzy: Administrator systemu, System

opis:

System umożliwia administratorowi przeglądanie listy dostępnych schematów danych. Lista zawiera szczegóły schematów, takie jak nazwa, opis i data ostatniej modyfikacji.

kroki:

2. **Administrator systemu** przechodzi do sekcji zarządzania schematami danych.
3. **System** pobiera z bazy danych listę schematów danych.
4. **System** wyświetla listę schematów danych administratorowi.

alt 1 Brak schematów danych:

1. **System** wykrywa, że w bazie danych nie ma dostępnych schematów.
2. **System** wyświetla komunikat informujący o braku schematów danych.

UC0031 Usuwanie schematów danych

nazwa: Usuwanie schematów danych

aktorzy: Administrator systemu, System, CKAN

opis:

System umożliwia administratorowi usunięcie istniejących schematów danych. Proces obejmuje wybór schematu do usunięcia, usunięcie go z systemu oraz synchronizację zmian z platformą CKAN.

kroki: 3. **Administrator systemu** wybiera schemat danych do usunięcia.

4. **Administrator systemu** zatwierdza usunięcie schematu.
5. **System** weryfikuje poprawność operacji.
6. **System** usuwa schemat danych z bazy danych.
7. **System** synchronizuje usunięcie schematu z platformą CKAN.
8. **System** wyświetla komunikat o pomyślnym usunięciu schematu danych.

alt 1 Schemat danych jest używany:

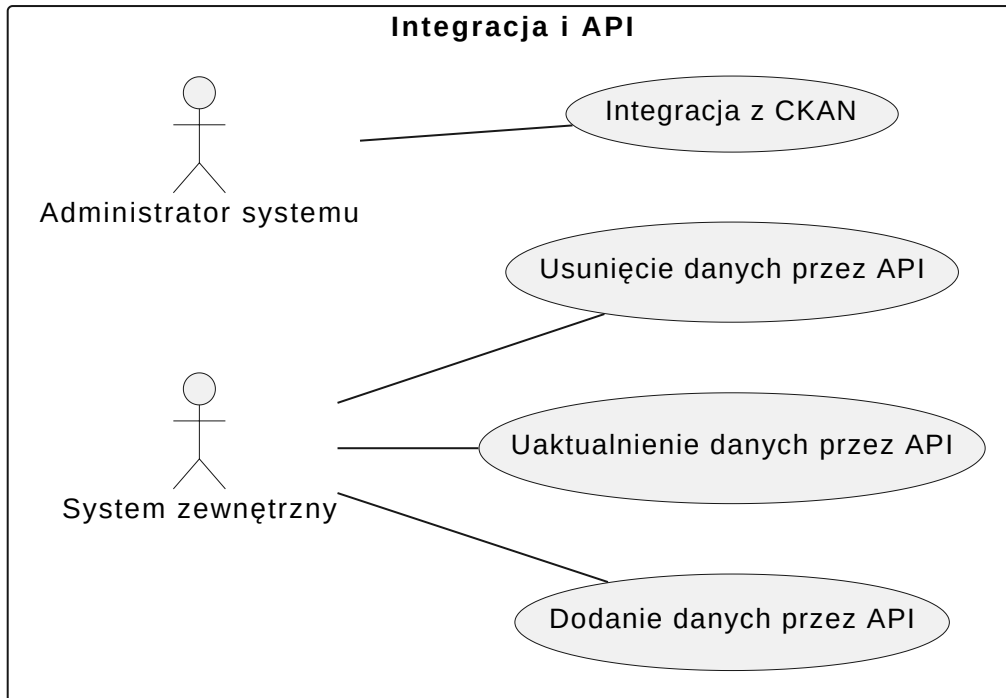
1. **System** wykrywa, że schemat danych jest powiązany z istniejącymi zbiorami danych.
2. **System** wyświetla komunikat o błędzie i uniemożliwia usunięcie schematu.

alt 2 Błąd synchronizacji z CKAN:

1. **System** napotyka błąd podczas synchronizacji z platformą CKAN.
2. **System** zapisuje zmiany lokalnie i wyświetla komunikat o błędzie synchronizacji.

4.1.5 Integracja i API

Diagram



Przypadki użycia

UC0032 Dodanie danych przez API

nazwa: Dodanie danych przez API

aktorzy: System zewnętrzny, System

opis:

System umożliwia zewnętrznym systemom dodawanie danych za pomocą API. Proces obejmuje przesłanie danych w odpowiednim formacie, ich walidację oraz zapisanie w systemie.

kroki:

1. **System zewnętrzny** wysyła żądanie dodania danych do API.
2. **System** odbiera żądanie i weryfikuje poprawność przesłanych danych.
3. **System** zapisuje dane w bazie danych.
4. **System** zwraca odpowiedź o pomyślnym dodaniu danych.

alt 1 Niepoprawne dane:

1. **System** wykrywa, że przesłane dane są niepoprawne.
2. **System** zwraca odpowiedź z informacją o błędzie i szczegółami problemu.

UC0033 Uaktualnienie danych przez API

nazwa: Uaktualnienie danych przez API

aktorzy: System zewnętrzny, System

opis:

System umożliwia zewnętrznym systemom aktualizację istniejących danych za pomocą API. Proces obejmuje przesłanie danych do aktualizacji, ich walidację oraz zapisanie zmian w systemie.

kroki:

1. **System zewnętrzny** wysyła żądanie aktualizacji danych do API.
2. **System** odbiera żądanie i weryfikuje poprawność przesłanych danych.
3. **System** aktualizuje dane w bazie danych.
4. **System** zwraca odpowiedź o pomyślnym uaktualnieniu danych.

alt 1 Niepoprawne dane:

1. **System** wykrywa, że przesłane dane są niepoprawne.
2. **System** zwraca odpowiedź z informacją o błędzie i szczegółami problemu.

UC0034 Usunięcie danych przez API

nazwa: Usunięcie danych przez API

aktorzy: System zewnętrzny, System

opis:

System umożliwia zewnętrznym systemom usuwanie danych za pomocą API. Proces obejmuje przesłanie identyfikatora danych do usunięcia, weryfikację żądania oraz usunięcie danych z systemu.

kroki:

1. **System zewnętrzny** wysyła żądanie usunięcia danych do API.
2. **System** odbiera żądanie i weryfikuje poprawność przesłanych danych.
3. **System** usuwa dane z bazy danych.
4. **System** zwraca odpowiedź o pomyślnym usunięciu danych.

alt 1 Dane nie istnieją:

1. **System** wykrywa, że dane do usunięcia nie istnieją w bazie danych.
2. **System** zwraca odpowiedź z informacją o błędzie.

UC0035 Integracja z CKAN

nazwa: Integracja z CKAN

aktorzy: Administrator systemu, System, CKAN

opis:

System umożliwia integrację z platformą CKAN w celu synchronizacji danych. Proces obejmuje konfigurację integracji, przesyłanie danych do CKAN oraz odbieranie danych z CKAN.

kroki: 2. **Administrator systemu** przechodzi do sekcji konfiguracji integracji z CKAN.

3. **Administrator systemu** wprowadza dane konfiguracyjne. 4. **Administrator systemu** zatwierdza konfigurację.

5. **System** nawiązuje połączenie z platformą CKAN i weryfikuje poprawność konfiguracji.

6. **System** synchronizuje dane z platformą CKAN. 7. **System** wyświetla komunikat o pomyślnym zakończeniu integracji.

alt 1 Błąd konfiguracji:

1. **System** wykrywa, że dane konfiguracyjne są niepoprawne.
2. **System** wyświetla komunikat o błędzie i prosi administratora o poprawienie konfiguracji.

alt 2 Błąd synchronizacji:

1. **System** napotyka błąd podczas synchronizacji z CKAN.
2. **System** wyświetla komunikat o błędzie..

4.1.6 Raportowanie i analiza

Diagram



Przypadki użycia

UC6001 Generowanie raportów dot. bezpieczeństwa

nazwa: Generowanie raportów dotyczących bezpieczeństwa

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia generowanie raportów dotyczących bezpieczeństwa danych, zawierających informacje o incydentach, nieautoryzowanym dostępie i innych zagrożeniach.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Generuj raport bezpieczeństwa".
2. **System** wyświetla dostępne kryteria (np. okres, typ incydentu).
3. **Klient/Użytkownik danych** wybiera kryteria i zatwierdza wybór.
4. **System** generuje raport na podstawie wybranych kryteriów.
5. **System** zapisuje raport i udostępnia go użytkownikowi.

alt 1 – Błąd generowania raportu:

1. **System** wykrywa problem podczas generowania raportu.
2. **System** wyświetla komunikat o błędzie i sugeruje ponowną próbę.

UC6002 Analiza raportów dot. bezpieczeństwa

nazwa: Analiza raportów dotyczących bezpieczeństwa

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia analizę wygenerowanych raportów dotyczących bezpieczeństwa, co pozwala na identyfikację trendów i potencjalnych zagrożeń.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Analizuj raport bezpieczeństwa".
2. **System** wyświetla listę dostępnych raportów bezpieczeństwa.
3. **Klient/Użytkownik danych** wybiera raport do analizy.
4. **System** prezentuje szczegółowe dane raportu (np. wykresy, tabele, wskaźniki).
5. **Klient/Użytkownik danych** dokonuje analizy raportu.

alt 1 – Brak raportów:

1. **System** wyświetla komunikat o braku dostępnych raportów bezpieczeństwa.
-

UC6003 Generowanie raportów dot. wydajności

nazwa: Generowanie raportów dotyczących wydajności

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia tworzenie raportów dotyczących wydajności systemu, prezentujących statystyki takie jak obciążenie serwera, czas reakcji i inne kluczowe metryki.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Generuj raport wydajności".
2. **System** prezentuje dostępne kryteria raportowania (np. okres, wybrane metryki).
3. **Klient/Użytkownik danych** wybiera kryteria i zatwierdza wybór.
4. **System** generuje raport na podstawie podanych kryteriów.
5. **System** zapisuje i udostępnia raport użytkownikowi.

alt 1 – Błąd generowania raportu:

1. **System** wyświetla komunikat o błędzie i sugeruje ponowną próbę generacji.
-

UC6004 Analiza raportów dot. wydajności

nazwa: Analiza raportów dotyczących wydajności

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia analizę wygenerowanych raportów wydajności, co pozwala na ocenę działania systemu oraz identyfikację potencjalnych usprawnień.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Analizuj raport wydajności".
2. **System** prezentuje listę dostępnych raportów wydajności.
3. **Klient/Użytkownik danych** wybiera raport do analizy.
4. **System** wyświetla szczegóły raportu (wykresy, tabele, statystyki).
5. **Klient/Użytkownik danych** dokonuje analizy raportu.

alt 1 – Raport nie zawiera danych:

1. **System** wyświetla komunikat informujący, że raport nie zawiera wystarczających danych do analizy.
-

UC6005 Raportowanie o ruchu

nazwa: Raportowanie o ruchu

aktorzy: Klient/Użytkownik danych, System

opis:

System generuje raporty dotyczące ruchu w systemie, prezentując statystyki takie jak liczba użytkowników, liczba operacji oraz inne metryki aktywności.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Raportuj ruch".
2. **System** zbiera dane o ruchu w określonym okresie.
3. **System** generuje raport zawierający kluczowe metryki.
4. **System** wyświetla raport użytkownikowi.

alt 1 – Brak danych o ruchu:

1. **System** wyświetla komunikat o braku dostępnych danych do raportowania.
-

UC6006 Lista wygenerowanych raportów

nazwa: Lista wygenerowanych raportów

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia przeglądanie historii wygenerowanych raportów, wraz z metadanymi (data, typ raportu) oraz opcją pobrania lub dalszej analizy.

kroki:

1. **Klient/Użytkownik danych** wybiera opcję "Lista raportów".
2. **System** pobiera listę wszystkich wygenerowanych raportów z bazy.
3. **System** wyświetla listę raportów z informacjami o dacie, typie oraz innymi metadanymi.
4. **Klient/Użytkownik danych** wybiera raport do pobrania lub analizy.

alt 1 – Brak wygenerowanych raportów:

1. **System** wyświetla komunikat informujący o braku wygenerowanych raportów.

4.1.7 System ocen i opinii**Diagram**



Przypadki użycia

UC7001 Wystawienie oceny zbioru danych

nazwa: Wystawienie oceny zbioru danych

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia użytkownikowi wystawienie oceny dla wybranego zbioru danych, co wpływa na jego ogólną ocenę oraz umożliwia innym użytkownikom zapoznanie się z opiniami.

kroki:

1. **Klient/Użytkownik danych** wybiera zbiór danych, który chce ocenić.
2. **System** wyświetla interfejs umożliwiający wystawienie oceny (np. skala od 1 do 5 gwiazdek) oraz dodanie komentarza.
3. **Klient/Użytkownik danych** wybiera wartość oceny oraz wpisuje opcjonalny komentarz.
4. **Klient/Użytkownik danych** zatwierdza wystawienie oceny.
5. **System** zapisuje ocenę i aktualizuje średnią ocen zbioru.
6. **System** wyświetla komunikat potwierdzający pomyślne wystawienie oceny.

alt 1 – Nieprawidłowa wartość oceny:

1. **System** wykrywa nieprawidłowy format lub zakres oceny.
 2. **System** wyświetla komunikat o konieczności poprawienia wprowadzonych danych.
-

UC7002 Aktualizacja oceny zbioru danych

nazwa: Aktualizacja oceny zbioru danych

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia użytkownikowi aktualizację wcześniej wystawionej oceny, w tym zmianę wartości oraz komentarza.

kroki:

1. **Klient/Użytkownik danych** loguje się i wybiera zbiór danych, dla którego chce zaktualizować ocenę.
2. **System** wyświetla dotychczasową ocenę i komentarz.
3. **Klient/Użytkownik danych** modyfikuje wartość oceny oraz/lub komentarz.
4. **Klient/Użytkownik danych** zatwierdza zmiany.
5. **System** zapisuje zaktualizowaną ocenę oraz przelicza średnią ocen zbioru.
6. **System** wyświetla komunikat potwierdzający pomyślną aktualizację oceny.

alt 1 – Błąd podczas aktualizacji:

1. **System** wykrywa błąd przy zapisie zmienionej oceny.
 2. **System** wyświetla komunikat o błędzie i prosi o ponowne wprowadzenie zmian.
-

UC7003 Wyświetlenie historii ocen

nazwa: Wyświetlenie historii ocen

aktorzy: Klient/Użytkownik danych, System

opis:

System umożliwia użytkownikowi przeglądanie historii ocen wystawionych dla danego zbioru danych, wraz z datami i komentarzami.

kroki:

1. **Klient/Użytkownik danych** wybiera zbiór danych, dla którego chce zobaczyć historię ocen.
2. **System** pobiera historię ocen z bazy danych.
3. **System** generuje listę zawierającą szczegóły każdej oceny (data, wartość, komentarz).
4. **System** wyświetla historię ocen użytkownikowi.

alt 1 – Brak historii ocen:

1. **System** wyświetla komunikat informujący, że dla wybranego zbioru nie ma zarejestrowanych ocen.

4.1.8 Zarządzanie wersjonowaniem i archiwizacją

Diagram



Przypadki użycia

UC0045 Tworzenie kopii zapasowych

nazwa: Tworzenie kopii zapasowych

aktorzy: Administrator systemu, System

opis: System umożliwia Administratorowi systemu tworzenie kopii zapasowych danych przechowywanych w systemie. Kopie zapasowe są tworzone w celu zabezpieczenia danych przed ich utratą w wyniku awarii systemu, błędów użytkownika lub innych nieprzewidzianych okoliczności. Proces tworzenia kopii zapasowej obejmuje zapisywanie danych w bezpiecznej lokalizacji, gdzie będą mogły zostać przywrócone w przypadku potrzeby.

kroki:

1. **Administrator systemu** wybiera opcję tworzenia kopii zapasowej.
2. **System** inicjuje proces tworzenia kopii zapasowej danych.
3. **System** wybiera dane, które mają zostać uwzględnione w kopii zapasowej (np. wszystkie dane, wybrane zbiory danych, konfiguracje systemowe).

4. **System** tworzy kopię zapasową i zapisuje ją w wybranej lokalizacji (np. chmura, zewnętrzny serwer, nośnik fizyczny).
5. **System** wyświetla komunikat potwierdzający pomyślne zakończenie tworzenia kopii zapasowej.

alt 1 Brak dostępnej przestrzeni na kopię zapasową:

1. **Administrator systemu** wybiera opcję tworzenia kopii zapasowej.
2. **System** sprawdza dostępność przestrzeni do zapisania kopii zapasowej.
3. **System** wykrywa brak wystarczającej przestrzeni do zapisania kopii zapasowej.
4. **System** wyświetla komunikat o błędzie i informuje administratora o braku miejsca na kopię zapasową.

alt 2 Błąd podczas tworzenia kopii zapasowej:

1. **Administrator systemu** wybiera opcję tworzenia kopii zapasowej.
2. **System** napotyka błąd podczas próby zapisania kopii zapasowej (np. problemy z dostępem do nośnika danych, błąd w procesie zapisu).
3. **System** wyświetla komunikat o błędzie i informuje administratora o problemach z tworzeniem kopii zapasowej.