

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI

PROJEKT Z BAZ DANYCH – Etap III

Temat projektu: Baza danych miejscowości w Polsce.

Termin zajęć: Piątek, 11:15–13:00

AUTOR/AUTORZY:

Dawid Waligórski, 264015

Dominika Nurek, 260525

Joanna Zoglowek 264452

PROWADZĄCY ZAJĘCIA:

dr inż. Roman Ptak

Wrocław, 1.12.2023 r.

Spis treści

1. Cele projektowe.....	2
2. Analiza wymagań.....	2
2.1 Opis słowny systemu.....	2
2.2 Schemat logiczny systemu	2
2.3 Wymagania funkcjonalne	3
2.3.1 Odnoszące się do wszystkich użytkowników systemu.....	3
2.3.2 Odnoszące się wyłącznie do administratorów technicznych.....	3
2.3.3 Odnoszące się wyłącznie do administratorów merytorycznych.....	4
2.4 Wymagania niefunkcjonalne	5
2.4.1 Wymagania techniczne	5
2.4.2 Wymagania dotyczące bezpieczeństwa i spójności danych	5
2.4.3 Wymagania wydajnościowe	5
3. Projekt systemu	6
3.1 Projekt bazy danych	6
3.1.1 Model koncepcyjny	6
3.1.2 Model logiczny.....	7
3.1.3 Model fizyczny.....	8
3.1.4 Inne elementy bazy danych	9
3.1.5 Procedury składowane.....	13
3.1.5 Model bezpieczeństwa na poziomie bazy danych.....	21
3.2 Projekt aplikacji dostępowej.....	23
3.2.1 Modelowanie przypadków użycia.....	23
3.2.2 Model bezpieczeństwa na poziomie aplikacji dostępowej	24
3.2.3 Projekt interfejsu graficznego.....	25
4. Implementacja bazy danych	36
4.1 Szczegóły implementacji.....	36
4.1.1 Tworzenie tabel i definiowanie ograniczeń	36
4.1.2 Implementacja mechanizmów przetwarzania danych	36
4.1.3 Implementacja uprawnień i innych zabezpieczeń	36
4.2 Testy poprawności	36
4.2.1 Widoki	Błąd! Nie zdefiniowano zakładki.
4.2.2 Wyzwalacze.....	38
4.2.3 Procedury składowane i wyzwalacza	36

1. Cele projektowe

Celem projektu jest stworzenie bazy danych miejscowości w Polsce, współpracując przy tym z grupami projektowymi, których tematy także poruszają tematykę geografii i podziału administracyjnego Polski. Dostęp do bazy danych odbywać się będzie poprzez dedykowaną aplikację dostępową. Przyjmie ona formę encyklopedii miejscowości, zawierającej informacje turystyczne.

2. Analiza wymagań

2.1 Opis słowny systemu

Projektowany system bazodanowy będzie zdolny do obsługi najważniejszych danych na temat, interesujących z punktu widzenia turystów miejscowości, zlokalizowanych na terenie Rzeczypospolitej Polskiej. Ich liczbę szacuje się na około tysiąc.

Użytkowników projektowanego systemu bazodanowego możemy podzielić zasadniczo na trzy grupy: administratorów merytorycznych, administratorów technicznych oraz przeglądających. Przy czym warto pamiętać, że administrator (zarówno techniczny jak i merytoryczny) może być jednocześnie przeglądającym. Jeden użytkownik nie może jednak pełnić obowiązków administratora technicznego i merytorycznego jednocześnie. Łączną liczbę użytkowników, korzystających z systemu w jednym momencie szacuje się na około 50. Rolą administratorów jest zarządzanie dostępnymi danymi w bazie, które mogą dodawać, usuwać i modyfikować. Przeglądający mają wyłącznie możliwość przeglądania listy miejscowości oraz szczegółowych danych miejscowości.

Wszyscy użytkownicy systemu korzystają z desktopowej aplikacji dostępowej, która pełni rolę interfejsu między nimi a wewnętrzną bazą danych. Aplikacja ta powinna zawierać przyjazny w obsłudze, graficzny interfejs użytkownika. W celu zwiększenia wygody użytkowników powinna dawać również możliwość dowolnego filtrowania i sortowania listy miejscowości.

W zadaniu projektowym zakładana jest także możliwość integracji tworzonej bazy danych z bazami stworzonymi przez inne grupy. Taka integracja odbywać się będzie na poziomie aplikacji dostępowej, która będzie przystosowana do komunikacji z bazami danych innych grup projektowych. Komunikacja ta sprowadzać się będzie wyłącznie do wysyłania zapytań do tych baz i odpowiadania na zapytania od nich. Co za tym idzie, modyfikowanie danych w zewnętrznych bazach będzie niedozwolone. Niedozwolone będzie również dopuszczenie do zmodyfikowania wewnętrznej bazy danych przez polecenia pochodzące z zewnętrznych baz danych.

2.2 Schemat logiczny systemu



Rysunek 1: Schemat logiczny systemu, obrazujący komunikację między jego wewnętrznymi modułami oraz z zewnętrznymi aktorami (użytkownik, zewnętrzne systemy bazodanowe)

2.3 Wymagania funkcjonalne

2.3.1 Odnoszące się do wszystkich użytkowników systemu

1. Wszyscy użytkownicy rejestrują się w systemie. Zakładają w ten sposób osobiste konta, które posiadają unikalną nazwę (login), hasło logowania i rolę (administrator techniczny/merytoryczny lub przeglądający).
2. Wszyscy użytkownicy logują się na swoje konta podając nazwę użytkownika oraz hasło.
3. Wszyscy użytkownicy mogą usunąć swoje konto z systemu. Wyjątkiem od tej reguły jest sytuacja, gdy w systemie zarejestrowane jedno konto administratora technicznego. Wówczas użytkownik posiadający takie konto nie może go usunąć.
4. Wszyscy użytkownicy mają możliwość dodania miejscowości do własnej listy ulubionych miejscowości.
5. Wszyscy użytkownicy mają możliwość usunięcia miejscowości z własnej listy ulubionych miejscowości.
6. Wszyscy użytkownicy mają możliwość przeglądania własnej listy ulubionych miejscowości. Lista pozwala na wyświetlenie wyłącznie szczegółowych danych na temat miejscowości. Listę tę można filtrować po atrybutach miejscowości. Dodatkowo istnieje będzie możliwość sortowania listy względem atrybutów miejscowości.
7. Wszyscy użytkownicy mają możliwość przeglądania listy wszystkich miejscowości. Lista pozwala na wyświetlenie wyłącznie szczegółowych danych na temat miejscowości: nazwę, liczbę atrakcji turystycznych, przydział do jednostek administracyjnych, populację, informacji czy dana miejscowość została dodana do listy ulubionych przeglądającego listę użytkownika. Listę tę mogą filtrować po atrybutach miejscowości takich jak: nazwa, populacja, przydział do jednostek administracyjnych, typy dostępnych atrakcji, typy miejscowości, liczba dostępnych atrakcji. Dodatkowo istnieje będzie możliwość sortowania listy względem atrybutów miejscowości takich jak: nazwa, populacja, liczba dostępnych atrakcji.
8. Wszyscy użytkownicy mogą przeglądać szczegółowe informacje o dowolnej miejscowości i zlokalizowanych w niej atrakcjach po wybraniu jej z listy miejscowości lub listy ulubionych miejscowości. Do szczegółowych informacji zaliczają się: nazwa, opis, populacja, przydział do jednostek administracyjnych, współrzędne geograficzne, typy miejscowości, informacje o wszystkich atrakcjach turystycznych przypisanych do tej miejscowości, informacja czy dana miejscowość została dodana do listy ulubionych przeglądającego listę użytkownika.

2.3.2 Odnoszące się wyłącznie do administratorów technicznych

9. Administrator techniczny może przeglądać listę wszystkich użytkowników zarejestrowanych w systemie. Lista pozwala na wyświetlenie szczegółowych informacji na temat użytkownika: login, rola. Z listy administrator techniczny może wybrać użytkownika w celu zarządzania jego kontem.
10. Administrator techniczny może przeglądać szczegółowe informacje o dowolnym użytkowniku zarejestrowanym w systemie. Do szczegółowych informacji zaliczają się: login, rola, lista przypisanych do zarządzania regionów wraz przydzielonymi w nich uprawnieniami.
11. Administrator techniczny może zmienić role przypisane do kont użytkowników. Przykładowo zmienić rolę użytkownika-przeglądającego na administratora technicznego.
12. Administrator techniczny może usuwać konta wszystkich użytkowników poza tymi, którzy są administratorami technicznymi.
13. Administrator techniczny może uprawnnić wskazanego administratora merytorycznego do dodawania miejscowości/atrakcji, usuwania miejscowości/atrakcji i modyfikacji danych miejscowości/atrakcji, znajdujących się w obrębie wskazanego województwa.
14. Administrator techniczny może odbierać uprawnienia wskazanemu administratorowi merytorycznemu do dodawania miejscowości/atrakcji, usuwania miejscowości/atrakcji i modyfikacji danych miejscowości/atrakcji, znajdujących się w obrębie wskazanego województwa.

2.3.3 Odnoszące się wyłącznie do administratorów merytorycznych

15. Administrator merytoryczny może przeglądać listę przydzielonych swojemu kontu uprawnień. Widoczne są przy tym dla niego typ uprawnień oraz województwo, w którym one obowiązują.
16. Administrator merytoryczny może dodać do bazy nowe miejscowości. Przy czym takie miejscowości muszą znajdować się w obrębie wyłącznie tych województw, do których dany administrator merytoryczny otrzymał uprawnienia. Wprowadza przy tym ich dane takie jak: nazwa miejscowości, opis miejscowości, populacja, przydział do jednostek administracyjnych, współrzędne geograficzne, typy miejscowości, listę atrakcji turystycznych.
17. Administrator merytoryczny może modyfikować dane miejscowości takie jak: nazwa miejscowości, opis miejscowości, populacja, przydział do jednostek administracyjnych, współrzędne geograficzne, typy miejscowości, listę atrakcji turystycznych. Przy czym takie miejscowości muszą znajdować się w obrębie wyłącznie tych województw, do których dany administrator merytoryczny otrzymał uprawnienia.
18. Administrator merytoryczny może usuwać miejscowości z bazy. Przy czym takie miejscowości muszą znajdować się w obrębie wyłącznie tych województw, do których dany administrator merytoryczny otrzymał uprawnienia.
19. Administrator merytoryczny może dodać do bazy informacje na temat atrakcji turystycznej, która dostępna jest w danej miejscowości. Na te informacje składają się: miejscowość, w której zlokalizowana jest atrakcja, nazwa atrakcji, opis atrakcji, typy atrakcji, zdjęcie. Przy czym taka atrakcja przypisana może być tylko do miejscowości, znajdujących się w obrębie wyłącznie tych województw, do których dany administrator merytoryczny otrzymał uprawnienia.
20. Administrator merytoryczny może modyfikować dane atrakcji turystycznych takie jak: miejscowość, w której zlokalizowana jest atrakcja, nazwa atrakcji, opis atrakcji, typy atrakcji, zdjęcie. Przy czym taka atrakcja przypisana może być tylko do miejscowości, znajdujących się w obrębie wyłącznie tych województw, do których dany administrator merytoryczny otrzymał uprawnienia.
21. Administrator merytoryczny może usuwać z bazy informacje na temat atrakcji turystycznej. Przy czym taka atrakcja przypisana może być tylko do miejscowości, znajdujących się w obrębie wyłącznie tych województw, do których dany administrator merytoryczny otrzymał uprawnienia.

2.4 Wymagania niefunkcjonalne

2.4.1 Wymagania techniczne

1. System jest uruchamialny na popularnych systemach operacyjnych (Windows, Linux).
2. System jest dostępny dla użytkowników w postaci aplikacji desktopowej z graficznym interfejsem użytkownika, wykonanej w technologii Java (Swing Framework).
3. System posiada przyjazny i prosty w obsłudze interfejs graficzny.
4. Dane turystyczne na temat miejscowości oraz dane dotyczące kont użytkowników przechowywane są w wewnętrznej bazie danych MariaDB. Do zamodelowania wspomnianej bazy posłuży narzędzie Visual Paradigm.
5. System jest przystosowany do komunikacji z zewnętrznymi systemami bazodanowymi na poziomie warstwy aplikacji dostępowej.

2.4.2 Wymagania dotyczące bezpieczeństwa i spójności danych

6. System chroni dane przed modyfikacjami zleconymi przez nieuprawnionych do tego aktorów (użytkowników-przeglądających, administratorów merytorycznych bez uprawnień w obrębie danej jednostki administracyjnej).
7. System zapewnia bezpieczeństwo danych użytkowników poprzez system osobistych kont oraz mechanizm logowania przy wykorzystaniu loginu i hasła.
8. Pierwszemu, założonemu w systemie kontu przypisywana jest rola administratora technicznego.
9. W systemie musi być zarejestrowane co najmniej jedno konto administratora technicznego.
10. Każdemu nowo założonemu kontu (poza pierwszym) przypisana zostaje rola przeglądającego.
11. Każde nowo utworzone konto administratora merytorycznego nie posiada uprawnień do usuwania, dodawania i modyfikacji danych na temat miejscowości lub atrakcji w żadnej z rejestrowanych przez bazę danych jednostce administracyjnej.
12. System zapewnia mechanizmy dbające o poprawność wprowadzanych przez administratorów danych.

2.4.3 Wymagania wydajnościowe

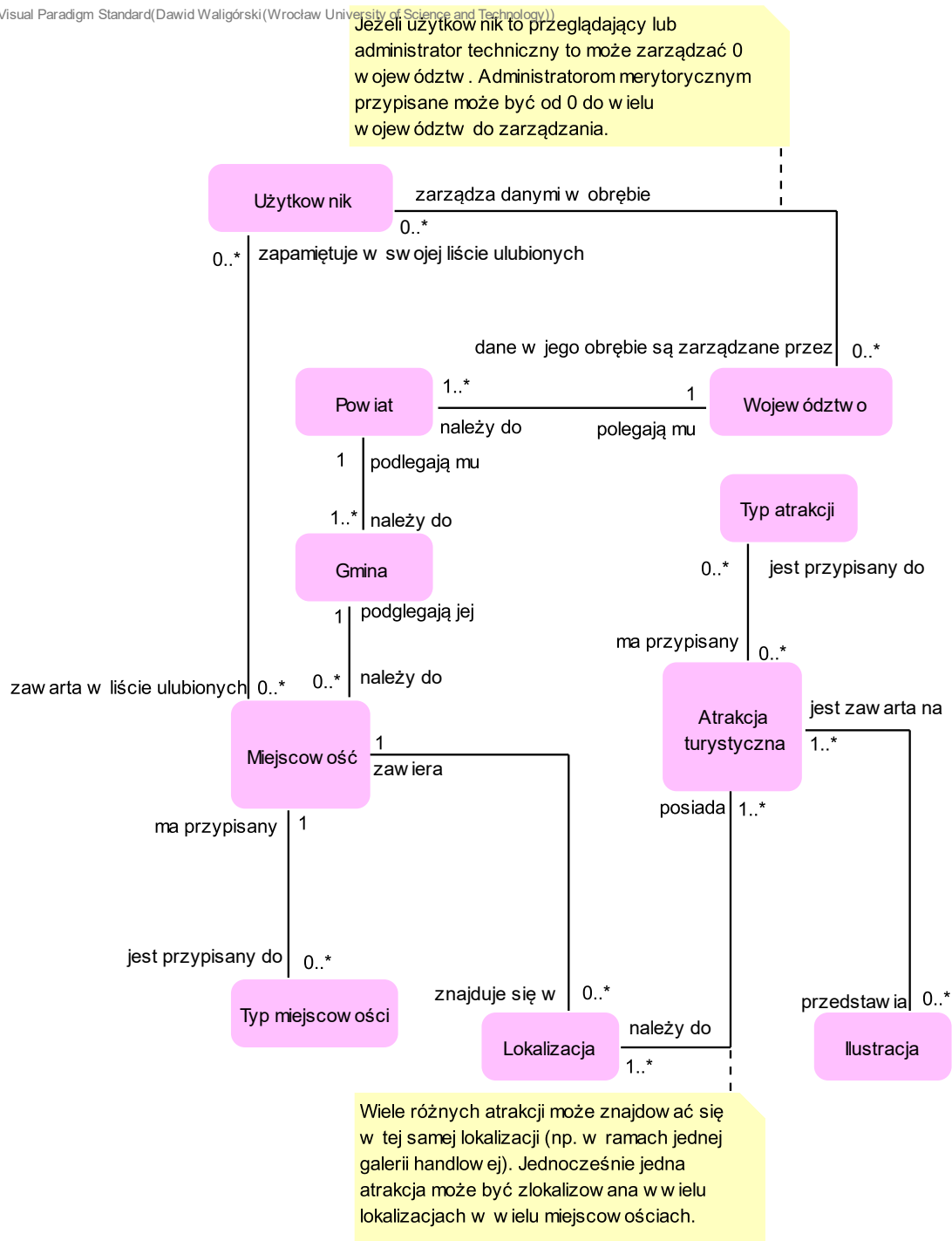
13. System jest w stanie obsługiwać do 50 użytkowników jednocześnie.
14. System powinien być w stanie efektywnie obsłużyć dane na temat około tysiąca miejscowości i dowolnej liczby ich atrakcji. Średnią liczbę atrakcji przypadającą na jedną miejscowość oszacowano na 10. Stąd liczbę elementów (atrakcji) przechowywanych w bazie danych szacuje się od 10 do 15 tysięcy.

3. Projekt systemu

3.1 Projekt bazy danych

3.1.1 Model konceptualny

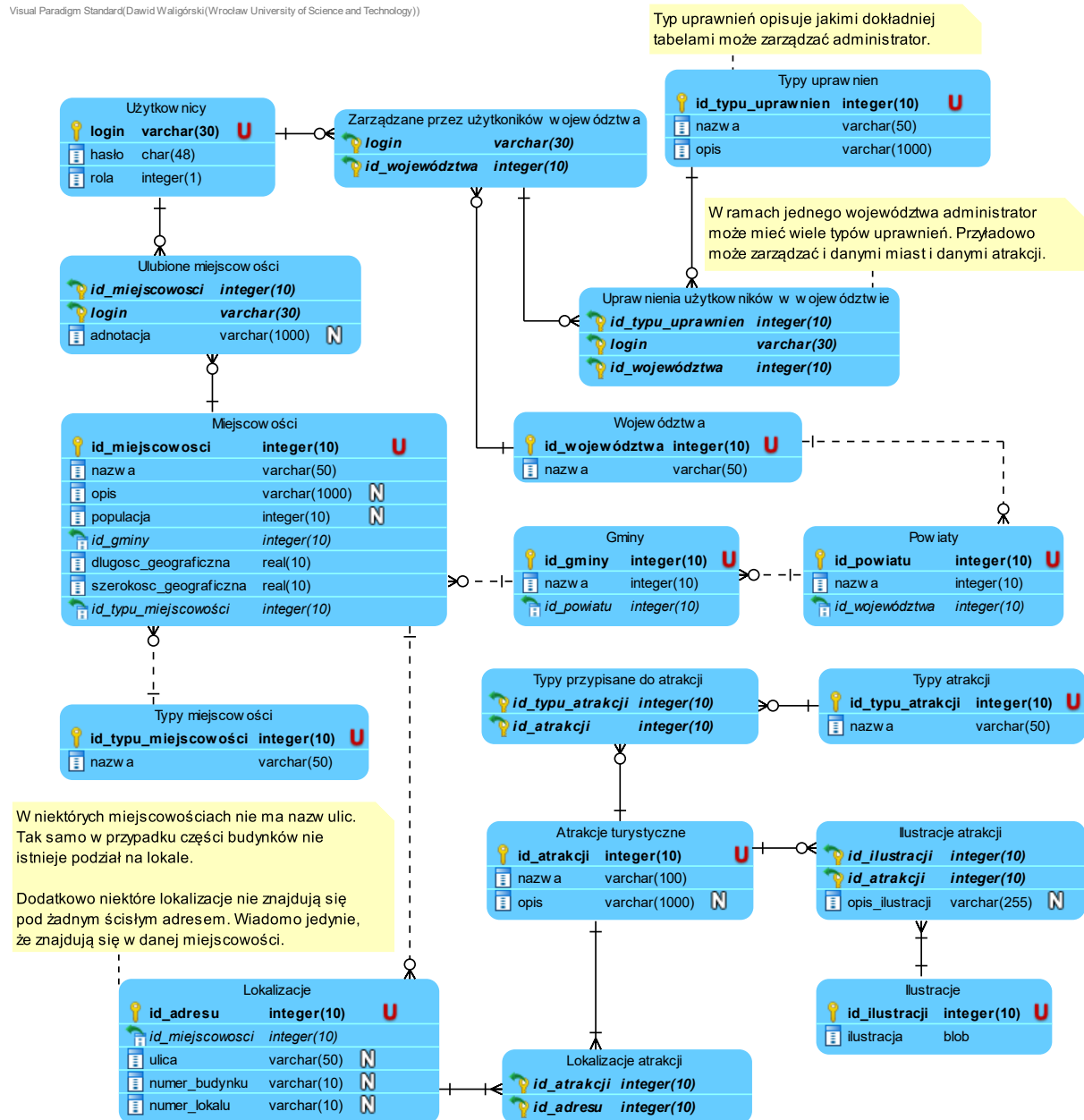
Visual Paradigm Standard (Dawid Waligórski (Wrocław University of Science and Technology))



Rysunek 2: Diagram przedstawiający model konceptualny projektowanej bazy danych.

3.1.2 Model logiczny

Visual Paradigm Standard(Dawid Waligórski(Wrocław University of Science and Technology))



Rysunek 3: Diagram przedstawiający model logiczny projektowanej bazy danych.

Visual Paradigm Standard (Dawid Waligórski/Wrocław University of Science and Technology)



3.1.4 Inne elementy bazy danych

3.1.4.1 Widoki

Tabela 1: Spis widoków w projektowanej bazie danych cz. 1


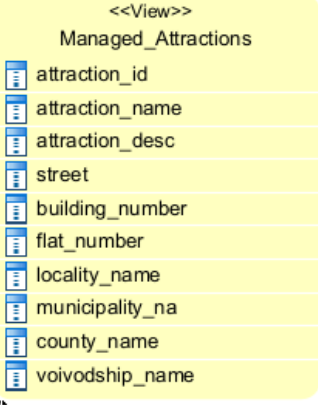
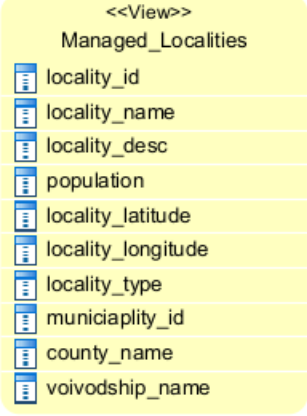
Nazwa	Rysunek	Opis
Full_Localities_Data	 <pre> <<View>> Full_Localities_Data locality_id locality_name locality_desc population locality_latitude locality_longitude locality_type municipality_name county_name voivodship_name </pre>	Pełne dane na temat wszystkich miejscowości z bazy danych.
Managed_Attractions	 <pre> <<View>> Managed_Attractions attraction_id attraction_name attraction_desc street building_number flat_number locality_name municipality_name county_name voivodship_name </pre>	Atrakcje, którymi może zarządzać aktualnie zalogowany użytkownik.
Managed_Localities	 <pre> <<View>> Managed_Localities locality_id locality_name locality_desc population locality_latitude locality_longitude locality_type municipality_id county_name voivodship_name </pre>	Miejscowości zarządzane przez aktualnie zalogowanego użytkownika

Tabela 2: Spis widoków w projektowanej bazie danych cz. 2

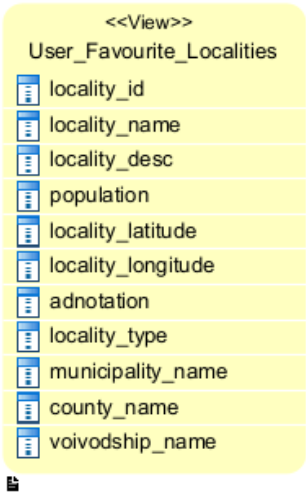
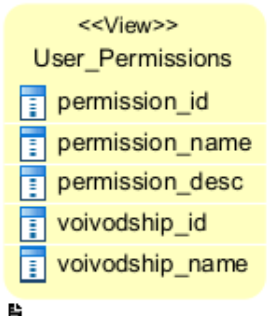
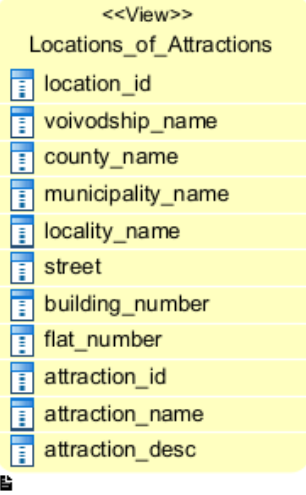
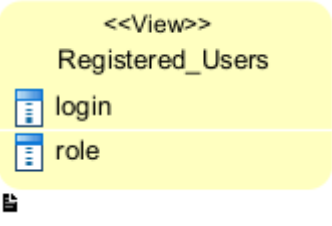
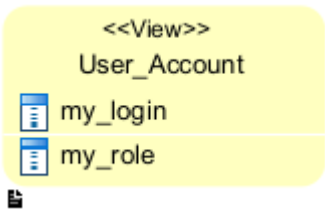
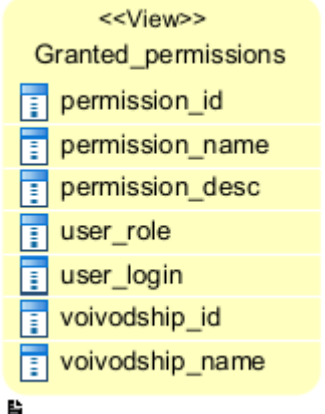
Nazwa	Rysunek	Opis
User_Favourite_Localities	 <pre> <<View>> User_Favourite_Localities locality_id locality_name locality_desc population locality_latitude locality_longitude adnotation locality_type municipality_name county_name voivodship_name </pre>	Dane miejscowości znajdujących się na liście ulubionych aktualnie zalogowanego użytkownika.
User_Permissions	 <pre> <<View>> User_Permissions permission_id permission_name permission_desc voivodship_id voivodship_name </pre>	Dane na temat wszystkich uprawnień posiadanych przez aktualnie zalogowanego użytkownika.
Locations_of_Attractions	 <pre> <<View>> Locations_of_Attractions location_id voivodship_name county_name municipality_name locality_name street building_number flat_number attraction_id attraction_name attraction_desc </pre>	Zawiera wszystkie lokalizacje atrakcji zapisane w bazie danych wraz z ich pełnymi danymi.
Registered_Users	 <pre> <<View>> Registered_Users login role </pre>	Zawiera spis wszystkich użytkowników z pominięciem wrażliwych danych.

Tabela 3: Spis widoków w projektowanej bazie danych cz. 3

Nazwa	Rysunek	Opis
User_Account	 <pre> <<View>> User_Account my_login my_role </pre>	Dane konta obecnie zalogowanego użytkownika.
Granted_permissions	 <pre> <<View>> Granted_permissions permission_id permission_name permission_desc user_role user_login voivodship_id voivodship_name </pre>	Zawiera spis wszystkich nadanych użytkownikom uprawnień.

3.1.4.2 Wyzwalacze

Tabela 4: Spis wyzwalaczy w projektowanej bazie danych.

Warunek wyzwalania	Opis
Before Delete On Users	<ul style="list-style-type: none"> Zapobiega usunięciu jedyne go konta o roli administratora technicznego.
Before Update On Users	<ul style="list-style-type: none"> Zapobiega zmianie roli jedyne go konta o roli administratora technicznego
After Delete On Locations	<ul style="list-style-type: none"> Usunięcie wszystkich atrakcji, dla których była to jedyna lokalizacja.
After Delete On Attractions_locations	<ul style="list-style-type: none"> Usunięcie wszystkich lokalizacji, które nie są przypisane do żadnej atrakcji. Usunięcie wszystkich atrakcji, które nie są przypisane do żadnej lokalizacji.
After Delete On Figures_Containing_Attractions	<ul style="list-style-type: none"> Usunięcie wszystkich ilustracji, które nie są przypisane do żadnej atrakcji.
After Delete On Users_Permissions_In_Voivodships	<ul style="list-style-type: none"> Usunięcie użytkownikom uprawnień do zarządzania wszystkimi województwami, w których nie mają przypisanych uprawnień żadnego typu (tabela Voivodships Administrated by Users).

3.1.4.3 Sekwencje

W używanej w ramach projektu bazie danych MariaDB nie istnieją tradycyjnie rozumiane sekwencje, służące do generowania kluczy głównych. W ich miejsce użyty zostanie mechanizm automatycznego generowania kluczy głównych (AUTO INCREMENT).

3.1.4.4 Indeksy

Tabela 5: Spis indeksów w projektowanej bazie danych.

Tabela	Kolumna
Localities	name
Localities	municipality_id
Localities	locality_type_id
Administrative_units	superior_administrative_unit
Locations	locality_id
Attractions	name
Users Permissions In Voivodships	login
Voivodships Administrated By Users	login
Attraction types	name
Favourite localities	login
Types assigned to attractiosn	attraction_id
Locality types	name
Figures containing attractions	attraction_id

3.1.5 Procedury składowane

3.1.5.1 unassign_permission_from_user

Służy do odebrania użytkownikom uprawnień do edycji w województwach (tabela Users_Permissions_In_Voivodships). Na parametry procedury składają się:

- login (varchar) – login użytkownika, któremu odbierane są uprawnienia.
- voivodship_id (integer) – identyfikator województwa, w którym odebrane ma zostać uprawnienie.
- permission_id (integer) – identyfikator typu uprawnienia, które ma zostać odebrane.

Parametry voivodship_id oraz permission_id mogą przyjąć wartość NULL. Wpływa to znacząco na efekt działania procedury.

voivodship_id	permission_id	Efekt
NOT NULL	NOT NULL	Usunięcie wskazanemu użytkownikowi uprawnienia wskazanego typu we wskazanym regionie.
NOT NULL	NULL	Usunięcie wskazanemu użytkownikowi wszystkich uprawnień we wskazanym regionie.
NULL	NOT NULL	Wywołanie wyjątku.
NULL	NULL	Usunięcie wskazanemu użytkownikowi wszystkich uprawnień we wszystkich województwach. Używane np. w wypadku, gdy użytkownikowi odebrana zostanie rola administratora merytorycznego.

Procedura nie wykona się w wypadku, gdy wskazany użytkownik nie będzie posiadać roli administratora merytorycznego. Podobnie zdarzy się w wypadku, gdy wartości zadanych parametrów nie będą znajdować się w bazie danych.

3.1.5.2 unassign_figure_from_attraction

Służy do usunięcia wskazanej ilustracji ze zbioru ilustracji przypisanych do wskazanej atrakcji (tabela Figures_Containing_Attractions). Na parametry procedury składają się:

- figure_id (integer) – identyfikator ilustracji, której przypisanie do atrakcji chcemy usunąć.
- attraction_id (integer) – identyfikator atrakcji, do której przypisanie do ilustracji chcemy usunąć.

Procedura nie wykona się w wypadku nieistnienia powiązania między wskazanymi elementami. Podobnie zdarzy się w wypadku zadania w którymkolwiek z parametrów wartości NULL.

3.1.5.3 unassign_attraction_from_locality

Służy do usunięcia wskazanej atrakcji z listy atrakcji znajdujących się we wskazanej miejscowości (tabela Attraction_locations).

- attraction_id (integer) – identyfikator usuwanej z miejscowości atrakcji.
- locality_id (integer) – identyfikator miejscowości, z której usuwana jest atrakcja.

Procedura nie wykona się w wypadku nieistnienia powiązania między wskazanymi elementami. Podobnie zdarzy się w wypadku zadania w którymkolwiek z parametrów wartości NULL.

3.1.5.4 unassign_type_from_attraction

Służy do usunięcia wskazanego typu atrakcji z listy typów przypisanych wskazanej atrakcji (tabela `Types_Assigned_To_Attractions`). Na parametry procedury składają się:

- `attraction_type_id` (integer) – identyfikator typu atrakcji, który ma zostać usunięty z listy typów wskazanej atrakcji.
- `attraction_id` (integer) – identyfikator atrakcji, której odbierany jest wskazany typ.

Parametr `attraction_type_id` może przyjąć wartość `NULL`. Ma to wpływ na działanie procedury.

<code>attraction_type_id</code>	Efekt
NOT NULL	Usunięcie wskazanego typu atrakcji z listy typów atrakcji wskazanej atrakcji.
NULL	Wyczyszczenie listy typów atrakcji wskazanej atrakcji.

Procedura nie wykona się w wypadku nieistnienia powiązania między wskazanymi elementami. Podobnie zdarzy się w wypadku zadania w parametrze `attraction_id` wartości `NULL`.

3.1.5.6 assign_permission_to_user

Służy do nadania wskazanemu użytkownikowi uprawnienia o wskazanym typie we wskazanym województwie (tabela `Users_Permissions_In_Voivodships` oraz w wypadku, gdy użytkownik nie miał nadawanych jeszcze żadnych uprawnień we wskazanym województwie również tabela `Voivodships_Administrated_By_Users`). Na parametry procedury składają się:

- `voivodship_id` (integer) – identyfikator województwa, w którym nadane ma zostać uprawnienie.
- `login` (varchar) – login użytkownika, któremu nadawane jest uprawnienie.
- `permission_id` (integer) – identyfikator typu uprawnienia nadawanego użytkownikowi.

Procedura nie wykona się w wypadku, gdy wskazany użytkownik nie będzie posiadać roli administratora metrytorycznego. Podobnie zdarzy się w wypadku, gdy wartości zadanych parametrów nie będą znajdować się w bazie danych. Żaden z parametrów nie może również przyjąć wartości `NULL`.

3.1.5.7 assign_attraction_to_locality

Służy do przypisania wskazanej atrakcji wskazanej lokalizacji w ramach miejscowości (tabela `Attractions_locations` oraz w wypadku, gdy lokalizacja o wskazanych parametrach nie znajduje się w bazie również tabela `Locations`). Na parametry procedury składają się:

- `attraction_id` (integer) – identyfikator atrakcji, która zostanie przypisana do lokalizacji.
- `locality_id` (integer) – identyfikator miejscowości, do której przypisana zostanie atrakcja.
- `street` (varchar) – nazwa ulicy, przy której znajduje się atrakcja.
- `building_number` (varchar) – numer budynku, w którym znajduje się atrakcja.
- `flat_number` (varchar) – numer lokalu, w którym znajduje się atrakcja.

Procedura nie wykona się w przypadku, gdy dowolny z parametrów `attraction_id` lub `locality_id` nie będzie znajdować się w bazie bądź przyjmie wartość `NULL`. Pozostałe parametry mogą przyjąć wartość `NULL`. Dodatkowo wskazana lokalizacja musi znajdować się w miejscowości, która należy do województwa zarządzanego przez wywołującego procedurę użytkownika.

3.1.5.8 assign_figure_to_attraction

Służy do przypisania wskazanej ilustracji wraz z podpisem do wskazanej atrakcji (tabela `Figures_Containing_Attractions`). Na parametry procedury składają się:

- `figure` (integer) – identyfikator ilustracji, która ma zostać przypisana do atrakcji.
- `attraction_id` (integer) – identyfikator atrakcji, do której ma zostać przypisana ilustracja.
- `caption` (varchar) – podpis ilustracji, która ma zostać przypisana do atrakcji.

Procedura nie wykona się w przypadku, gdy co najmniej dowolny z parametrów `figure_id` lub `locality_id` nie będzie znajdować się w bazie bądź przyjmie wartość `NULL`. Pozostałe parametry mogą przyjąć wartość `NULL`. Dodatkowo atrakcja, do której przypisywana jest ilustracja musi być zlokalizowana w województwie, zarządzanym przez wywołującego procedurę użytkownika.

3.1.5.9 assign_type_to_attraction

Służy do przypisania wskazanego typu atrakcji do wskazanej atrakcji (tabela `Types_Assigned_To_Attractions`). Na parametry procedury składają się:

- `type_id` (integer) – identyfikator typu atrakcji, który ma zostać przypisany.
- `attraction_id` (integer) – identyfikator atrakcji, do której ma zostać przypisany typ atrakcji.

Procedura nie wykona się w przypadku, gdy dowolny z jej parametrów nie będzie znajdować się w bazie danych bądź przyjmie wartość `NULL`. Dodatkowo atrakcja, do której przypisywany jest typ musi być zlokalizowana w województwie, zarządzanym przez wywołującego procedurę użytkownika. On sam zaś musi mieć uprawnienia do zarządzania atrakcjami w tymże województwie.

3.1.5.10 del_locality

Służy do usunięcia wskazanej miejscowości z bazy danych (tabela `Localities`). Na parametry procedury składają się:

- `locality_id` (integer) – identyfikator miejscowości, która ma zostać usunięta.

Procedura nie wykona się w przypadku, gdy jej parametr nie będzie znajdować się w bazie danych bądź przyjmie wartość `NULL`. Dodatkowo usuwana miejscowość musi być zlokalizowana w województwie, zarządzanym przez wywołującego procedurę użytkownika.

3.1.5.11 del_locality_from_fav_list

Służy do usunięcia wskazanej miejscowości z listy ulubionych wywołującego ją użytkownika (tabela `Favourite_Localities`). Na parametry procedury składają się:

- `locality_id` (integer) – identyfikator miejscowości, która ma być usunięta z listy ulubionych.

Procedura nie wykona się w przypadku, gdy jej parametr nie będzie znajdować się w liście ulubionych lub przyjmie wartość `NULL`.

3.1.5.12 del_user

Służy do usunięcia wskazanego konta użytkownika z systemu (tabela `Users`). Na parametry procedury składają się:

- `login` (varchar) – login konta, które ma zostać usunięte.

Procedura nie wykona się w przypadku, gdy jej parametr nie będzie znajdować się w bazie użytkowników. W wypadku podania w parametrze wartości `NULL` nastąpi usunięcie konta aktualnie zalogowanego użytkownika. Jeżeli wywołujący ją użytkownik nie ma roli administratora technicznego, a wskazany zostanie login innego użytkownika, to procedura nie wykona się.

3.1.5.13 del_attraction

Służy do usunięcia wskazanej atrakcji z bazy danych (tabela Localities). Na parametry procedury składają się:

- `attraction_id` (integer) – identyfikator atrakcji, która ma zostać usunięta.

Procedura nie wykona się w przypadku, gdy jej parametr nie będzie znajdować się w bazie danych lub przyjmie wartość NULL. Dodatkowo usuwana atrakcja musi być zlokalizowana w województwie, zarządzanym przez wywołującego procedurę użytkownika.

3.1.5.14 modify_locality

Służy do modyfikowania danych wskazanej miejscowości (tabela Localities). Na parametry procedury składają się:

- `locality_id` (integer) – identyfikator miejscowości, która ma zostać zmodyfikowana.
- `name` (varchar) – nowa nazwa miejscowości.
- `desc` (varchar) – nowy opis miejscowości.
- `pop` (integer) – nowa populacja miejscowości.
- `type` (integer) – identyfikator typu miejscowości.
- `municipality_id` (integer) – identyfikator gminy, w której znajduje się miejscowość.
- `lat` (real) – szerokość geograficzna miejscowości.
- `long` (real) – długość geograficzna miejscowości.

Procedura nie wykona się w wypadku, gdy jej parametry `locality_id`, `type` i `municipality_id` nie będą znajdować się w bazie danych. Wszystkie inne parametry mogą przyjąć wartość NULL, co oznaczać będzie, że odpowiadające im kolumny mają zostać niezmienione. Co ważne modyfikowana miejscowość przed wywołaniem procedury musi znajdować się w województwie, którym zarządza wywołujący procedurę użytkownik.

3.1.5.15 modify_attraction

Służy do modyfikowania danych wskazanej atrakcji (tabela Attractions). Na parametry procedury składają się:

- `attraction_id` (integer) – identyfikator atrakcji, która ma zostać zmodyfikowana.
- `name` (varchar) – nowa nazwa atrakcji.
- `desc` (varchar) – nowy opis atrakcji.

Procedura nie wykona się w przypadku, gdy jej parametr `attraction_id` nie będzie znajdować się w bazie danych. Wszystkie inne parametry mogą przyjąć wartość NULL, co oznaczać będzie, że odpowiadające im kolumny mają zostać niezmienione. Co ważne modyfikowana atrakcja musi znajdować się w województwie, którym zarządza wywołujący procedurę użytkownik.

3.1.5.16 modify_user_role

Służy do modyfikowania roli wskazanego użytkownika (tabela Users). Na parametry procedury składają się:

- `login` (varchar) – login użytkownika, którego rola ma być zmieniona.
- `role` (varchar) – identyfikator określający nową rolę użytkownika.

Procedura nie wykona się w przypadku, gdy jej parametr `login` nie będzie znajdować się w bazie danych bądź przyjmą wartość NULL. Dodatkowo parametr `role` powinien przyjmować jedną z 3 dozwolonych wartości odpowiadających rolom: przeglądającego, administratora technicznego, administratora merytorycznego.

3.1.5.17 modify_figure_caption

Służy do modyfikowania podpisu wskazanej ilustracji, która jest przypisana do wskazanej atrakcji (tabela Figures_Containing_Attractions). Na parametry procedury składają się:

- `figure_id` (integer) – identyfikator ilustracji, której podpis ma zostać zmodyfikowany.
- `attraction_id` (integer) – identyfikator atrakcji, do której przypisana jest ilustracja, której podpis ma być zmodyfikowany.
- `caption` (varchar) – nowy napis, który ma zostać przypisany do obrazka.

Procedura nie wykona się w wypadku, gdy jej wskazana atrakcja nie jest powiązana ze wskazanym obrazkiem. Nie wykona się również, gdy którykolwiek z jej parametrów przyjmie wartość NULL. Co ważne obrazek musi być przypisany do atrakcji, znajdującej się w województwie zarządzanym przez użytkownika wywołującego procedurę.

3.1.5.18 add_new_locality

Służy do dodawania nowej miejscowości do bazy danych. Na parametry procedury składają się:

- `name` (varchar) – nazwa miejscowości.
- `desc` (varchar) – opis miejscowości.
- `pop` (integer) – populacja miejscowości.
- `municipality_id` (integer) – identyfikator gminy, w której znajduje się miejscowość.
- `lat` (real) – szerokość geograficzna miejscowości.
- `long` (real) – długość geograficzna miejscowości.
- `type` (integer) – identyfikator typu miejscowości

Procedura nie wykona się w przypadku, gdy jej parametr `municipality_id` nie będzie znajdować się w bazie danych, przyjmie wartość NULL lub nie będzie przypisany do jednostki administracyjnej będącej gminą. Istnieć w bazie danych musi również parametr `type`. Wszystkie inne parametry poza nazwą miejscowości mogą przyjąć wartość NULL. Co ważne miejscowość musi znajdować się w województwie zarządzanym przez użytkownika wywołującego procedurę.

3.1.5.19 add_new_attraction

Służy do dodawania nowej atrakcji do bazy danych. Na parametry procedury składają się:

- `name` (varchar) – nazwa atrakcji.
- `desc` (varchar) – opis atrakcji.
- `locality_id` (integer) – identyfikator miejscowości, w której znajduje się atrakcja.
- `street` (varchar) – nazwa ulicy, na której znajduje się atrakcja.
- `buildig_number` (varchar) – numer budynku, w którym znajduje się atrakcja.
- `flat_number` (varchar) – numer lokalu, w którym znajduje się atrakcja.

Procedura nie wykona się w przypadku, gdy jej parametr `locality_id` nie będzie znajdować się w bazie danych lub przyjmie wartość NULL. Wartości NULL nie może przyjąć również parametr `name`. Dodatkowo procedura na podstawie parametrów tworzy wewnątrz miejscowości nową lokalizację atrakcji bądź przypisuje jej już istniejącą lokalizację (o ile ma ona takie same dane jak te zadane w parametrach).

3.1.5.20 add_locality_to_fav_list

Służy do dodania wskazanej miejscowości do listy ulubionych wywołującego ją użytkownika. Jednocześnie zapisuje adnotację do miejscowości dodaną przez tegoż użytkownika. Na parametry procedury składają się:

- `locality_id` (integer) – identyfikator miejscowości, która ma zostać dodana do ulubionych.
- `adnotation` (varchar) – adnotacja dodawana do miejscowości.

Procedura nie wykona się w wypadku, gdy jej parametr `locality_id` nie znajduje się w bazie danych lub przyjmie wartość NULL.

3.1.5.21 get_localities_number_of_attractions

Służy do pobierania liczby atrakcji wskazanej miejscowości. Na jej parametry składają się:

- `locality_id` (integer) – identyfikator miejscowości, której liczbę atrakcji chcemy wyznaczyć.

Procedura zwróci wartość -1 jeżeli jej parametr `locality_id` nie znajduje się w bazie danych lub przyjmie wartość NULL.

3.1.5.22 get_locations_from_locality

Służy do pobierania listy lokalizacji znajdujących się we wskazanej miejscowości. Na jej parametry składają się:

- `locality_id` (integer) – identyfikator miejscowości, której listę lokalizacji pobieramy.

Procedura nie wykona się jeżeli jej parametr `locality_id` nie znajduje się w bazie danych lub przyjmie wartość NULL. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.23 get_voivodships_managed_by_user

Służy do pobierania listy województw zarządzanych przez wskazanego użytkownika. Na parametry procedury składają się:

- `login` (varchar) – login użytkownika, którego listę zarządzanych województw pobieramy.

Procedura nie wykona się jeżeli jej parametr `login` nie znajduje się w bazie danych. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.24 get_user_permissions_in_voivodship

Służy do pobierania listy uprawnień posiadanych przez wskazanego użytkownika we wskazanym województwie. Na parametry procedury składają się:

- `login` (varchar) – login użytkownika, którego listę uprawnień pobieramy.
- `voivodship_id` (integer) – identyfikator województwa, którego mają dotyczyć się uprawnienia.

Procedura nie wykona się jeżeli jej parametry nie znajdują się w bazie danych bądź przyjmą wartość NULL. Nie wykona się również, gdy `voivodship_id` nie będzie identyfikatorem jednostki administracyjnej będącej województwem. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.25 get_attractions_in_locality

Służy do pobierania listy wszystkich atrakcji ze wskazanej miejscowości. Na parametry procedury składają się:

- `locality_id` (integer) – identyfikator miejscowości, z której atrakcje chcemy wyliczyć.

Procedura nie wykona się jeżeli jej parametr nie znajduje się w bazie danych bądź przyjmie wartość NULL. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.26 get_figures_assigned_to_attraction

Służy do pobierania listy wszystkich ilustracji przypisanych do wskazanej atrakcji. Na parametry procedury składają się:

- `attraction_id` (integer) – identyfikator atrakcji, której ilustracje chcemy pobrać.

Procedura nie wykona się jeżeli jej parametr nie znajduje się w bazie danych bądź przyjmie wartość NULL. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.27 get_types_assigned_to_attraction

Służy do pobierania listy wszystkich typów przypisanych do wskazanej atrakcji. Na parametry procedury składają się:

- `attraction_id` (integer) – identyfikator atrakcji, której typy chcemy pobrać.

Procedura nie wykona się jeżeli jej parametr nie znajduje się w bazie danych bądź przyjmie wartość NULL. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.28 get_counties_from_voivodship

Służy do pobierania listy wszystkich powiatów podlegających pod wskazane województwo. Na parametry procedury składają się:

- `voivodship_id` (integer) – identyfikator województwa, którego podległe powiaty chcemy pobrać.

Procedura nie wykona się jeżeli jej parametr nie znajduje się w bazie danych bądź przyjmie wartość NULL. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.29 get_municipalities_from_county

Służy do pobierania listy wszystkich gmin podlegających pod wskazany powiat. Na parametry procedury składają się:

- `county_id` (integer) – identyfikator powiatu, którego podległe gminy chcemy pobrać.

Procedura nie wykona się jeżeli jej parametr nie znajduje się w bazie danych bądź przyjmie wartość NULL. Pobrane dane zapisane są w tymczasowej tabeli (TEMPORARY TABLE) o nazwie `return_table`.

3.1.5.30 register_user

Służy do rejestrowania nowego użytkownika o wskazanym loginie i hasle w systemie (generuje wpisy w tabeli przechowujących użytkowników zarówno serwera bazodanowego jak i samej bazy danych). Na parametry procedury składają się:

- `login` (varchar) – login użytkownika, który ma zostać zarejestrowany.
- `password` (char) – hasło użytkownika, który ma zostać zarejestrowany.

Procedura nie wykona się jeżeli jej parametr `login` znajduje się już w bazie danych. Dodatkowo żaden z jej parametrów nie może przyjąć wartości NULL. Procedura zajmuje się przypisaniem odpowiedniej roli do konta użytkownika (pierwsze konto w systemie przyjmie rolę administratora

technicznego, pozostałe przeglądającego). Ponadto procedura przeprowadzi proces rejestracji konta użytkownika w serwerze bazodanowym i nadaniu mu odpowiednich uprawnień.

3.1.5 Model bezpieczeństwa na poziomie bazy danych

Model bezpieczeństwa na poziomie bazy danych oparty jest na podziale użytkowników na grupy: przeglądających (wszyscy użytkownicy), administratorów technicznych, administratorów merytorycznych. Każda z wymienionych ról upoważnia użytkownika do innych ingerencji w dane zapisanych w bazie. Dodatkowo w przypadku administratorów merytorycznych zakres prerogatyw dookreślany jest przez nadane im przez administratorów technicznych uprawnienia edycji danych w województwach.

Tabela 6: Spis prerogatyw użytkowników bazy danych w zależności od roli (cz. 1)

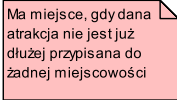
Rola	Przywileje
Przeglądający (Viewer)	<ul style="list-style-type: none"> • SELECT z Full_Localities_Data • SELECT z User_Favourite_Localities • SELECT z User_Permissions • SELECT z Locations_of_Attractions • SELECT z User_Account • EXECUTE procedurę del_locality_from_fav_list • EXECUTE procedurę del_user • EXECUTE procedurę add_locality_to_fav_list • EXECUTE procedurę get_localities_number_of_attractions • EXECUTE procedurę get_locations_from_locality • EXECUTE procedurę get_attractions_in_locality • EXECUTE procedurę get_figures_assigned_to_attraction • EXECUTE procedurę get_types_assigned_to_attraction • EXECUTE procedurę get_counties_from_voivodship • EXECUTE procedurę get_municipalities_from_county • SELECT z Administrative_unit • SELECT z Permissions • SELECT z Locality_Types • SELECT z Figures • SELECT z Attraction_Types
Administrator merytoryczny (Meritorical_Admin)	<ul style="list-style-type: none"> • SELECT z Managed_Localities • SELECT z Managed_Attractions • INSERT do Figures • DELETE z Figures • EXECUTE procedurę del_locality • EXECUTE procedurę modify_locality • EXECUTE procedurę add_new_locality • EXECUTE procedurę unassign_figure_from_attraction • EXECUTE procedurę unassign_attraction_from_locality • EXECUTE procedurę unassign_type_from_attraction • EXECUTE procedurę assign_attraction_to_locality • EXECUTE procedurę assign_figure_to_attraction • EXECUTE procedurę assign_type_to_attraction • EXECUTE procedurę del_attraction • EXECUTE procedurę modify_attraction • EXECUTE procedurę modify_figure_caption • EXECUTE procedurę add_new_attraction
Administrator techniczny (Technical_Admin)	<ul style="list-style-type: none"> • SELECT z Registered_Users • SELECT z Granted_permissions • EXECUTE procedurę unassign_permission_from_user • EXECUTE procedurę assign_permission_to_user

	<ul style="list-style-type: none"> • EXECUTE procedurę <code>modify_user_role</code> • EXECUTE procedurę <code>get_voivodships_managed_by_user</code> EXECUTE procedurę <code>get_user_permissions_in_voivodship</code>
--	---

Baza danych posiada również zbiór tabel słownikowych: `Administrative_units`, `Permissions`, `Locality_types`, `Attraction_types`. Są one uzupełniane wszystkimi niezbędnymi danymi przez zespół projektujący bazę danych (o kontach typu `root`) przed oddaniem jej od użytku. Tabela z kontami użytkowników również jest uzupełniana przez użytkowników o kontach typu `root` (później przez aplikację dostępową). Mają oni dostęp do procedury `register_user`.

3.2.1 Modelowanie przypadków użycia

Visual Paradigm Standard (Dominiika(Wroclaw University of Science and Technology))



Rysunek 5: Diagram przypadków użycia aplikacji dostępowej.

3.2.2 Model bezpieczeństwa na poziomie aplikacji dostępowej

Każdy użytkownik może zalogować się na osobiste konto. Każde konto posiada rolę: przeglądającego, administratora merytorycznego, administratora technicznego. Rola determinuje zakres możliwych do podjęcia w aplikacji dostępowej akcji. Przy czym role administratorów posiadają również prerogatywy roli przeglądającego. W wypadku administratorów merytorycznych wpływ na zakres możliwych do podjęcia akcji różni się w zależności od uprawnień do edycji nadanych im w poszczególnych województwach.

Tabela 7: Spis prerogatyw użytkowników aplikacji dostępowej w zależności od roli (cz. 1)

Rola	Prerogatywy
Przeglądający	<ul style="list-style-type: none">• Rejestracja• Logowanie• Wylogowanie• Przeglądanie listy miejscowości• Przeglądanie listy ulubionych miejscowości• Dodanie miejscowości do listy ulubionych• Usunięcie miejscowości z listy ulubionych• Przeglądanie danych miejscowości• Przeglądanie danych użytkownika (swojego konta)• Usunięcie konta użytkownika
Administrator merytoryczny z uprawnieniami do edycji miast w województwie	<ul style="list-style-type: none">• Edytowanie danych miejscowości (o ile jest w obrębie zarządzanego województwa)• Dodanie nowej miejscowości (o ile jest w obrębie zarządzanego województwa)• Usunięcie miejscowości (o ile jest w obrębie zarządzanego województwa)
Administrator merytoryczny z uprawnieniami do edycji atrakcji w województwie	<ul style="list-style-type: none">• Przypisanie atrakcji (o ile miejscowość jest w obrębie zarządzanego województwa)• Edytowanie danych atrakcji (o ile jest w obrębie zarządzanego województwa)• Dodanie nowej atrakcji (o ile jest w obrębie zarządzanego województwa)• Usunięcie atrakcji z miejscowości (o ile jest w obrębie zarządzanego województwa)• Usunięcie atrakcji (o ile jest w obrębie zarządzanego województwa)• Przypisanie adresu (o ile jest w obrębie zarządzanego województwa)• Dodanie nowego adresu (o ile jest w obrębie zarządzanego województwa)• Przypisanie ilustracji (o ile atrakcja jest w obrębie zarządzanego województwa)• Dodanie ilustracji (o ile atrakcja jest w obrębie zarządzanego województwa)

Tabela 8: Spis prerogatyw użytkowników aplikacji dostępowej w zależności od roli (cz. 2)

Rola	Prerogatywy
Administrator techniczny	<ul style="list-style-type: none"> • Przeglądanie listy użytkowników. • Usunięcie konta użytkownika (dowolnego) • Zmiana roli użytkownika • Odebranie uprawnień w regionie • Odebranie uprawnień do regionu • Nadanie uprawnień do regionu • Nadanie uprawnień w regionie

Dodatkowo aplikacja dostępowa czuwać będzie nad poprawnością wprowadzanych do systemu danych oraz szyfrowaniem haseł logowania użytkowników.

3.2.3 Projekt interfejsu graficznego

Interfejs graficzny aplikacji dostępowej składać się będzie z zestawu ekranów, umożliwiających skorzystanie z zamodelowanych przypadków użycia. Na wspomniany zestaw ekranów składają się:

- Ekran logowania i rejestracji
- Ekran główny
- Ekran przeglądania listy miejscowości/ulubionych miejscowości
- Ekran edytora miejscowości
- Ekran przeglądania szczegółów miejscowości
- Ekran przypisania adresu do atrakcji w miejscowości
- Ekran przypisania atrakcji do miejscowości
- Ekran edytora adresu w miejscowości
- Ekran edytora atrakcji
- Ekran nadawania uprawnień w regionie
- Ekran doboru typu uprawnienia
- Ekran doboru kryteriów filtrowania miejscowości
- Ekran doboru kryteriów filtrowania użytkowników
- Ekran listy użytkowników
- Ekran przypisania ilustracji do atrakcji
- Ekran doboru kryteriów sortowania miejscowości
- Ekran doboru sortowania użytkowników
- Ekran przeglądania szczegółów konta użytkownika

Wygląd poszczególnych ekranów zaprezentowany został w odpowiadających ich podrozdziałach.

3.2.3.1 Ekran logowania i rejestracji

Logowanie i rejestracja

Login:

Hasło:

Rysunek 6: Schemat ekranu logowania i rejestracji.

3.2.3.2 Ekran główny

login rola

[5] [4] [3] [2] [1]

ilustracja

Turystyczna baza danych miejscowości w Polsce

krótki opis bazy danych

Lista miejscowości Ulubione miejscowości

krótki opis procesu przeglądania miejscowości krótki opis procesu przeglądania ulubionych miejscowości

Otwórz Otwórz

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia uprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 7: Schemat ekranu głównego (widok domyślny)

login rola

[5] [4] [3] [2] [1]

ilustracja

Turystyczna baza danych miejscowości w Polsce

krótki opis bazy danych

Lista miejscowości Ulubione miejscowości Lista użytkowników

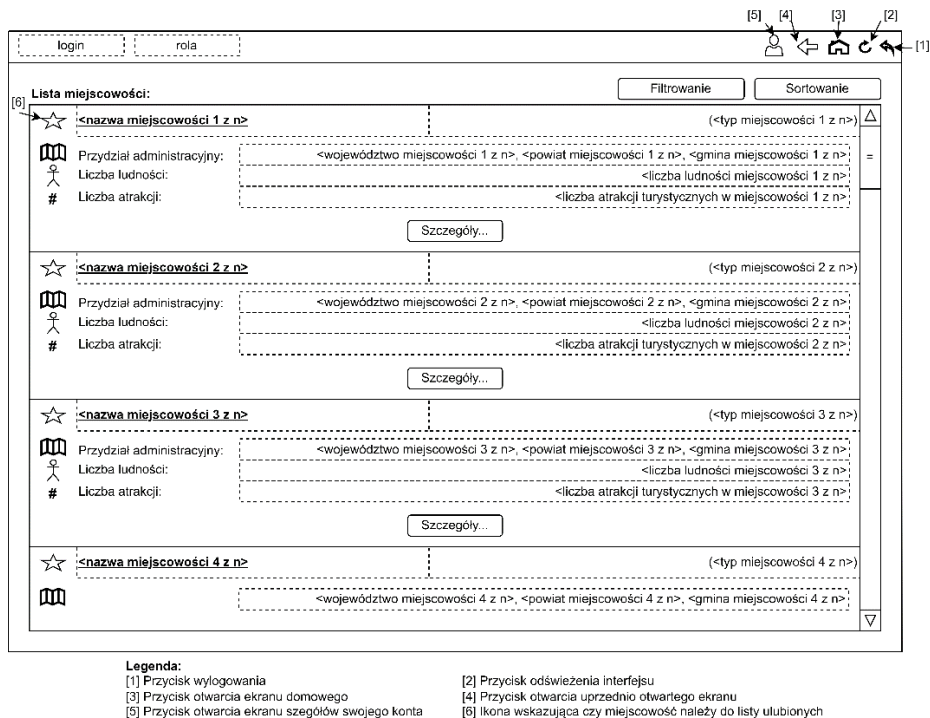
krótki opis procesu przeglądania miejscowości krótki opis procesu przeglądania ulubionych miejscowości krótki opis procesu przeglądania użytkowników

Otwórz Otwórz Otwórz

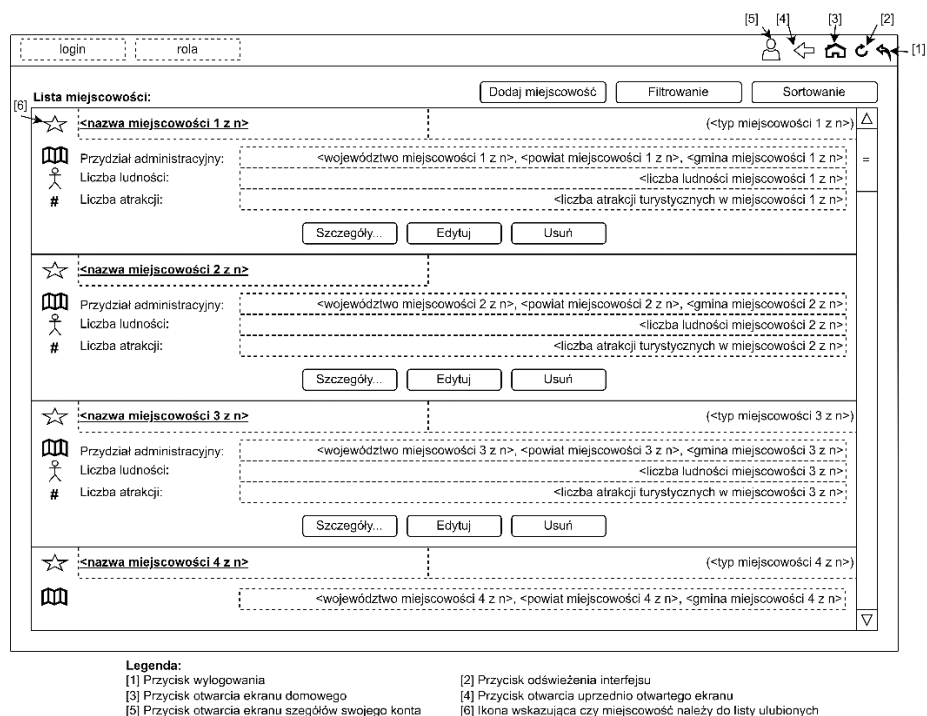
Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia uprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 8: Schemat ekranu głównego (widok administratora technicznego)

3.2.3.3 Ekran przeglądania listy miejscowości/ulubionych miejscowości



Rysunek 9: Schemat ekranu przeglądania listy miejscowości (widok domyślny)



Rysunek 10: Schemat ekranu przeglądania listy miejscowości (widok administratora merytorycznego)

3.2.3.4 Ekran edytora miejscowości

Edytor miejscowości:

Nazwa miejscowości:

Opis miejscowości:

Populacja:

Typ miejscowości:

Gmina

Długość i szerokość geograficzna:

Zatwierdź Anuluj

Legenda:
 [1] Przycisk wylogowania
 [2] Przycisk odświeżenia interfejsu
 [3] Przycisk otwarcia ekranu domowego
 [4] Przycisk otwarcia poprzednio otwartego ekranu
 [5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 11: Szablon ekranu edytora miejscowości

3.2.3.5 Ekran przeglądania szczegółów miejscowości

Przeglądanie szczegółów miejscowości:

<Nazwa miejscowości>

opis miejscowości

Dane systemowe:
 Populacja: <populacja>
 Typ miejscowości: <typ miejscowości>
 Województwo: <województwo>
 Powiat: <powiat>
 Gmina: <gmina>
 Dł. geograficzna: <długość geograficzna>
 Sz. geograficzna: <szerokość geograficzna>

Dane użytkownika:
 Adnotacja:
 Dodaj do ulubionych Usuń z ulubionych Zapisz adnotację

<nazwa atrakcji 1 z n> <adres atrakcji 1 z n w tej miejscowości>

opis atrakcji 1 z n

Typy atrakcji:
 wymienione po przecinku typy atrakcji 1 z n

Ilustracja:
 Ilustracja 1 z k atrakcji 1 z n

opis ilustracji 1 z k atrakcji 1 z n

Legenda:
 [1] Przycisk wylogowania
 [2] Przycisk odświeżenia interfejsu
 [3] Przycisk otwarcia ekranu domowego
 [4] Przycisk otwarcia poprzednio otwartego ekranu
 [5] Przycisk otwarcia ekranu szczegółów swojego konta
 [6] Ikona wskazująca czy miejscowość należy do listy ulubionych

Rysunek 12: Szablon ekranu przeglądania szczegółów miejscowości (widok domyślny)

login rola

[5] [4] [3] [2] [1]

[6] ☆ <Nazwa miejscowości> Przypisz atrakcję Edytuj Usuń

opis miejscowości

Dane systemowe:

Populacja: <populacja>

Typ miejscowości: <typ miejscowości>

Województwo: <województwo>

Powiat: <powiat>

Gmina: <gmina>

Dł. geograficzna: <długość geograficzna>

Sz. geograficzna: <szerokość geograficzna>

Dane użytkownika:

Adnotacja:

Edytowalne pole na adnotację dodawaną przez użytkownika do ulubionej miejscowości.

Dodaj do ulubionych Usuń z ulubionych Zapisz adnotację

<nazwa atrakcji 1 z n> <adres atrakcji 1 z n w tej miejscowości> Edytuj Usuń

opis atrakcji 1 z n

Typy atrakcji:

wymienione po przecinku typy atrakcji 1 z n

Ilustracje:

Ilustracja 1 z k atrakcji 1 z n

opis ilustracji 1 z k atrakcji 1 z n

Legenda:

[1] Przycisk wylogowania [2] Przycisk odświeżenia interfejsu

[3] Przycisk otwarcia ekranu domowego [4] Przycisk otwarcia uprzednio otwartego ekranu

[5] Przycisk otwarcia ekranu szczegółów swojego konta [6] Ikona wskazująca na obecność miejscowości na liście ulubionych

Rysunek 13: Szablon ekranu przeglądania szczegółów miejscowości (widok administratora merytorycznego)

3.2.3.6 Ekran przypisania adresu do atrakcji w miejscowości

login rola

[5] [4] [3] [2] [1]

Przypisanie adresu do atrakcji w miejscowości <nazwa miejscowości>:

Lista dostępnych adresów: Przypisz nowy adres

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

<ulica>, <numer budynku>, <numer lokalu> Przypisz

Anuluj

Legenda:

[1] Przycisk wylogowania [2] Przycisk odświeżenia interfejsu

[3] Przycisk otwarcia ekranu domowego [4] Przycisk otwarcia uprzednio otwartego ekranu

Rysunek 14: Szablon ekranu przypisania adresu do atrakcji w miejscowości.

3.2.3.7 Ekran przypisania atrakcji do miejscowości

login rola

[5] [4] [3] [2] [1]

Przypisanie atrakcji do miejscowości <nazwa miejscowości>

Lista dostępnych atrakcji: Przypisz nową atrakcję

<nazwa atrakcji 1 z 2>
Opis:
opis atrakcji 1 z 2
Przypisz

<nazwa atrakcji 2 z 2>
Opis:
opis atrakcji 2 z 2
Przypisz

Anuluj

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia upřednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 15: Szablon ekranu przypisania atrakcji do miejscowości.

3.2.3.8 Ekran edytora adresu w miejscowości

login rola

[5] [4] [3] [2] [1]

Edytor adresu w miejscowości <nazwa miejscowości>

Nazwa ulicy
edytowalne pole tekstowe na nazwę ulicy

Numer budynku
edytowalne pole tekstowe na numer budynku

Numer lokalu
edytowalne pole tekstowe na numer lokalu

Zatwierdź Anuluj

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia upřednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 16: Szablon ekranu edytora adresu w miejscowości

3.2.3.9 Ekran edytora atrakcji

login rola

[5] [4] [3] [2] [1]

Edytor atrakcji:

Nazwa atrakcji: edytowalne pole tekstowe na nazwę atrakcji

Opis atrakcji: edytowalne pole tekstowe na opis atrakcji

Typ atrakcji: lista wielokrotnego wyboru zawierająca typy atrakcji

Adresy atrakcji:

<miasteczko>, <ulica>, <numer budynku>, <numer lokalu> Przypisz nowy

<miasteczko>, <ulica>, <numer budynku>, <numer lokalu> Przypisz nowy

Ilustracje atrakcji: Przypisz ilustrację

Ilustracja: ilustracja 1 z n

Opis ilustracji: edytowalne pole tekstowe na opis ilustracji 1 z n

Usuń

Zatwierdź Anuluj

Legenda:

[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia poprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 17: Szablon ekranu edytora atrakcji

3.2.3.10 Ekran nadawania uprawnień w regionie

login rola

[5] [4] [3] [2] [1]

Edytor nadawania uprawnień w regionie użytkownikowi <login>

Województwo: lista jednokrotnego wyboru zawierająca województwa

Zatwierdź Anuluj

Legenda:

[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia poprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 18: Szablon ekranu nadawania uprawnień w regionie

3.2.3.11 Ekran doboru typu uprawnienia

login rola

[5] [4] [3] [2] [1]

Uprawnienia przypisane użytkownikowi <login> w województwie <nazwa województwa>

Uprawnienie
lista jednokrotnego wyboru zawierająca typy uprawnień

Opis typu uprawnień
nieedytowalne pole tekstowe zawierające opis wybranego typu uprawnień

Zatwierdź Anuluj

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia poprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 19: Szablon ekranu doboru typu uprawnienia

3.2.3.12 Ekran doboru kryteriów filtrowania miejscowości

login rola

[5] [4] [3] [2] [1]

Kryteria filtrowania miejscowości:

Nazwa miejscowości:
pole do wprowadzenia nazwy miejscowości (lub jej części) Reset

Liczba atrakcji:
min wprowadzenie dolnego ogr. max wprowadzenie górnego ogr. Reset

Populacja:
min wprowadzenie dolnego ogr. max wprowadzenie górnego ogr. Reset

Typ miejscowości:
lista wielokrotnego wyboru z dostępnych typów miejscowości Reset

Typy atrakcji:
lista wielokrotnego wyboru z dostępnych typów atrakcji Reset

Województwo
lista wielokrotnego wyboru z dostępnych województw Reset

Powiat
lista wielokrotnego wyboru z dostępnych powiatów Reset

Gmina
lista wielokrotnego wyboru z dostępnych gmin Reset

Zastosuj filtry Resetuj filtry

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia poprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 20: Szablon ekranu doboru kryteriów filtrowania miejscowości

3.2.3.13 Ekran doboru kryteriów filtrowania użytkowników

login rola

[5] [4] [3] [2] [1]

Kryteria filtrowania użytkowników:

Login:
pole do wprowadzenia loginu (lub jego części) Reset

Rola:
lista wielokrotnego wyboru z dostępnych ról Reset

Filtruj Resetuj filtry

Legenda:
[1] Przycisk wylogowania [2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego [4] Przycisk otwarcia uprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 21: Szablon ekranu doboru kryteriów filtrowania użytkowników

3.2.3.14 Ekran listy użytkowników

login rola

[5] [4] [3] [2] [1]

Lista użytkowników: Filtrowanie Sortowanie

[6]	<Login użytkownika 1 z 4>	<rola użytkownika 1 z 4>	Zarządzaj kontem
[6]	<Login użytkownika 2 z 4>	<rola użytkownika 2 z 4>	Zarządzaj kontem
[6]	<Login użytkownika 3 z 4>	<rola użytkownika 3 z 4>	Zarządzaj kontem
[6]	<Login użytkownika 4 z 4>	<rola użytkownika 4 z 4>	Zarządzaj kontem

Legenda:
[1] Przycisk wylogowania [2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego [4] Przycisk otwarcia uprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta [6] Ikona wskazująca na rolę konta

Rysunek 22: Szablon ekranu listy użytkowników

3.2.3.15 Ekran przypisania ilustracji do atrakcji

login rola

[5] [4] [3] [2] [1]

Przypisanie ilustracji do atrakcji:

Opis ilustracji:

edytowalne pole tekstowe zawierające opis ilustracji

Lista ilustracji: Przypisz nową ilustrację

ilustracja 1 z n Przypisz

ilustracja 2 z n Anuluj

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia uprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 23: Szablon ekranu przypisania ilustracji do atrakcji

3.2.3.16 Ekran doboru kryteriów sortowania miejscowości

login rola

[5] [4] [3] [2] [1]

Kryteria sortowania miejscowości:

Nazwa miejscowości:
lista jednokrotnego wyboru z dostępnych typów sortowań Reset

Liczba atrakcji:
lista jednokrotnego wyboru z dostępnych typów sortowań Reset

Populacja:
lista jednokrotnego wyboru z dostępnych typów sortowań Reset

Zastosuj sortowanie Resetuj sortowanie

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia uprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 24: Szablon ekranu doboru kryteriów sortowania miejscowości

3.2.3.17 Ekran doboru kryteriów sortowania użytkowników

login rola

[5] [4] [3] [2] [1]

Kryteria sortowania użytkowników:

Login
lista jednokrotnego wyboru z dostępnych typów sortowań ▼ Reset

Rola
lista jednokrotnego wyboru z dostępnych typów sortowań ▼ Reset

Sortuj Resetuj sortowanie

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia poprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 25: Ekran doboru kryteriów sortowania użytkowników

3.2.3.18 Ekran przeglądania szczegółów konta użytkownika

login rola

[5] [4] [3] [2] [1]

Dane użytkownika:
Login: <login>
Aktualnie przypisana rola: <rola>

Zarządzanie kontem użytkownika:
Usuń konto użytkownika

Zarządzane regiony:
<województwo>
Uprawnienia w regionie:
<nazwa typu uprawnień 1 z n>
Opis: opis typu uprawnień 1 z n

Legenda:
[1] Przycisk wylogowania
[2] Przycisk odświeżenia interfejsu
[3] Przycisk otwarcia ekranu domowego
[4] Przycisk otwarcia poprzednio otwartego ekranu
[5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 26: Szablon ekranu przeglądania szczegółów konta użytkownika (widok domyślny)

login **rola**

[5] [4] [3] [2] [1]

Dane użytkownika:
 Login: <login>
 Aktualnie przypisana rola: <rola>

Zarządzanie kontem użytkownika:
 Usuń konto użytkownika
 Zmiana przypisanej roli:
 lista jednokrotnego wyboru z listy dostępnych ról
 Zatwierdź

Zarządzane regiony:
 Nadaj uprawnienia do regionu

<województwo> Nadaj uprawnienie w regionie Odbierz uprawnienia do regionu

Uprawnienia w regionie:
 <nazwa uprawnienia 1 z n>
 Opis:
 opis typu uprawnień 1 z n
 Odbierz uprawnienie

Legenda:
 [1] Przycisk wylogowania
 [2] Przycisk odświeżenia interfejsu
 [3] Przycisk otwarcia ekranu domowego
 [4] Przycisk otwarcia poprzednio otwartego ekranu
 [5] Przycisk otwarcia ekranu szczegółów swojego konta

Rysunek 27: Szablon ekranu przeglądania szczegółów konta użytkownika (widok administratora technicznego)

4. Implementacja bazy danych

4.1 Szczegóły implementacji

4.1.1 Implementacja tabel oraz widoków

Do utworzenia tabeli służy zapytanie CREATE TABLE. Składnia zapytania: CREATE TABLE nazwa tabeli (kolumna 1 typ_danych, kolumna 2 typ_danych), gdzie kolumna definiuje nazwę kolumny (atrybutu), a typ_danych definiuje typ danych w danej kolumnie. Na typ danych możemy narzucić ograniczenia używając słowa kluczowego CHECK (..) i w nawiasie podając ograniczenia.

Ry

4.1.2 Implementacja innych elementów bazy danych

4.1.2.1 Implementacja wyzwalaczy

4.1.2.2 Implementacja procedur

4.1.3 Implementacja modelu bezpieczeństwa

4.2 Testy poprawności

4.2.1 Procedury składowane

4.2.1.1 Procedura add_new_attraction

Scenariusz 1:

Opis scenariusza:	Poprawne dodanie nowej atrakcji do bazy danych przez uprawnionego do tego administratora merytorycznego. Lokalizacja dodawanej atrakcji nie została uprzednio zapisana w bazie.
Zakładany rezultat:	Procedura wykona się. Do bazy danych dodana zostanie nowa atrakcja oraz jej lokalizacja.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 2:

Opis scenariusza:	Poprawne dodanie nowej atrakcji do bazy danych przez uprawnionego do tego administratora merytorycznego. Lokalizacja dodawanej atrakcji została uprzednio zapisana w bazie.
Zakładany rezultat:	Procedura wykona się. Do bazy danych dodana zostanie nowa atrakcja. Przypisany zostanie do niej istniejąca w bazie lokalizacja.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 3:

Opis scenariusza:	Poprawne dodanie nowej atrakcji do bazy danych przez nieuprawnionego do tego administratora merytorycznego.
Zakładany rezultat:	Procedura nie wykona się.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 4:

Opis scenariusza:	Dodanie nowej atrakcji do bazy danych przez uprawnionego do tego administratora merytorycznego do nieistniejącej w bazie miejscowości.
Zakładany rezultat:	Procedura nie wykona się.
Użyte polecenia:	
Uzyskany rezultat:	

4.2.1.2 Procedura assign_attraction_to_locality

Scenariusz 1:

Opis scenariusza:	Poprawne przypisanie atrakcji do miejscowości przez uprawnionego do tego administratora merytorycznego (do nieistniejącej w bazie lokalizacji).
Zakładany rezultat:	Procedura wykona się poprawnie. Nowa lokalizacja, znajdująca się we wskazanej miejscowości zostanie dodana do bazy. Zostanie do niej przypisana wskazana atrakcja.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 2:

Opis scenariusza:	Poprawne przypisanie atrakcji do miejscowości przez uprawnionego do tego administratora merytorycznego (do istniejącej w bazie lokalizacji).
Zakładany rezultat:	Procedura wykona się poprawnie. Atrakcja zostanie przypisana do istniejącej lokalizacji w bazie.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 3:

Opis scenariusza:	Poprawne przypisanie atrakcji do miejscowości przez nieuprawnionego do tego administratora merytorycznego.
Zakładany rezultat:	Procedura nie wykona się.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 4:

Opis scenariusza:	Poprawne przypisanie atrakcji do miejscowości przez nieuprawnionego do użytkownika (przeglądającego).
Zakładany rezultat:	Procedura nie wykona się.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 5:

Opis scenariusza:	Przypisanie atrakcji do nieistniejącej miejscowości.
Zakładany rezultat:	Procedura nie wykona się.
Użyte polecenia:	
Uzyskany rezultat:	

Scenariusz 6:

Opis scenariusza:	Przypisanie nieistniejącej atrakcji do miejscowości.
Zakładany rezultat:	Procedura nie wykona się.
Użyte polecenia:	
Uzyskany rezultat:	

4.2.2 Wyzwalacze