

**ZAGROŻENIA GEODYNAMICZNE I HYDROGEOLOGICZNE NA TERENACH
GÓRNICZYCH I POGÓRNICZYCH W GÓRNOŚLĄSKIM ZAGŁĘBIU WĘGLOWYM**

Zadanie 1.6. Opracowanie zintegrowanych map zagrożeń geodynamicznych
i hydrogeologicznych na terenach górniczych i pogórnich
w Górnośląskim Zagłębiu Węglowym

RAPORT KWARTALNY nr 02/2024

(za okres 01.01.2024 – 30.06.2024)

Zawartość raportu:

- I. Opracowanie szablonów i stylów wizualizacji danych monitoringowych.
- II. Geoprocessing testowych danych wsadowych i ich przetworzenie do formatu GeoPackage.
- III. Przygotowanie podstaw modelu wizualizacji danych przestrzennych w serwisie internetowym.

Katowice, lipiec 2024 r.

I. Opracowanie szablonów i stylów wizualizacji danych monitoringowych

1. Cel etapu

Celem tego etapu jest opracowanie spójnych i estetycznych szablonów oraz stylów wizualizacji danych monitoringowych, które ułatwią interpretację wyników i ich prezentację dla różnych grup interesariuszy. W szczególności należy uwzględnić potrzeby inżynierów, geologów, decydentów oraz społeczności lokalnej.

2. Wymagania dotyczące wizualizacji

1. Przystępność:

- Wizualizacje muszą być łatwe do zrozumienia dla użytkowników o różnym poziomie wiedzy technicznej.
- Należy unikać nadmiernej komplikacji i stosować jasne, intuicyjne symbole oraz kolory.

2. Czytelność:

- Dane muszą być prezentowane w sposób przejrzysty, z odpowiednimi etykietami i legendami.
- Ważne informacje powinny być wyróżnione i łatwo dostępne.

3. Spójność:

- Szablony i style wizualizacji muszą być spójne w całym projekcie, aby zapewnić jednolity wygląd i sposób interpretacji danych.
- Używane kolory, czcionki i formatowanie muszą być jednolite.

3. Narzędzia i oprogramowanie

- **GIS (Geographic Information System):** ArcGIS, QGIS
- **Oprogramowanie do analizy danych i wizualizacji:** MATLAB, R, Python (biblioteki matplotlib, seaborn, plotly)
- **Platformy do wizualizacji danych:** Tableau, Power BI
- **Oprogramowanie graficzne:** Adobe Illustrator, Inkscape

4. Kroki w opracowaniu szablonów i stylów

1. Analiza istniejących standardów i wymagań użytkowników:

- Przegląd standardów wizualizacji danych w geofizyce, geologii i hydrogeologii.
- Zbieranie informacji na temat preferencji i wymagań użytkowników końcowych.

2. Projektowanie podstawowych elementów wizualizacji:

- **Kolorystyka:** Dobór kolorów do reprezentacji różnych typów danych, takich jak warstwy geologiczne, poziomy wód gruntowych, anomalii geofizycznych.
- **Symbole i ikony:** Stworzenie zestawu symboli do reprezentacji punktów pomiarowych, stacji monitoringu, itp.
- **Czcionki i formatowanie:** Wybór czcionek i stylów tekstu dla tytułów, etykiet, legend i opisów.

3. Tworzenie szablonów wizualizacji:

- **Mapy tematyczne:** Szablony map prezentujących różne typy danych (geofizyczne, geologiczne, hydrogeologiczne).
- **Wykresy i diagramy:** Szablony wykresów czasowych, histogramów, diagramów kołowych itp.

- **Profile i przekroje:** Szablony dla profili sejsmicznych, geologicznych i hydrogeologicznych.
- 4. Prototypowanie i testowanie:**
- Tworzenie prototypów wizualizacji na podstawie danych testowych.
 - Przegląd prototypów przez zespół interesariuszy.
 - Wprowadzenie poprawek na podstawie feedbacku użytkowników.
- 5. Opracowanie instrukcji i dokumentacji:**
- Przygotowanie dokumentacji opisującej standardy wizualizacji, wytyczne dotyczące użycia szablonów i stylów oraz przykłady poprawnie wykonanych wizualizacji.
 - Stworzenie krótkich poradników i tutoriali w formie pisemnej lub wideo.

5. Przykłady wizualizacji

1. Mapy tematyczne:

- **Mapa gradientów anomalii geofizycznych:** Kolorowe kontury reprezentujące zmiany wartości anomalii.
- **Mapa warstw geologicznych:** Różne kolory i wzory reprezentujące poszczególne warstwy skalne.
- **Mapa izopiezometryczna:** Linie konturowe pokazujące poziom wód gruntowych.

2. Wykresy i diagramy:

- **Wykresy czasowe:** Wykresy pokazujące zmiany poziomów wód gruntowych lub anomalii geofizycznych w czasie.
- **Histogramy:** Rozkład wartości wybranych parametrów, takich jak przepuszczalność skał.
- **Diagramy kołowe:** Procentowy udział różnych typów skał w danym obszarze.

3. Profile i przekroje:

- **Przekrój sejsmiczny:** Profil pokazujący struktury podpowierzchniowe na podstawie danych sejsmicznych.
- **Profil geologiczny:** Warstwy geologiczne z opisem litologicznym.
- **Profil hydrogeologiczny:** Poziomy wód gruntowych oraz przepuszczalność warstw.

6. Wnioski i dalsze kroki

Po opracowaniu szablonów i stylów wizualizacji, kolejnym krokiem będzie ich wdrożenie w procesie tworzenia raportów i prezentacji danych monitoringowych. Konieczne będzie również przeszkolenie zespołu w zakresie korzystania z tych narzędzi oraz monitorowanie ich efektywności i wprowadzanie ewentualnych usprawnień na podstawie feedbacku użytkowników.

II. Geoprocessing testowych danych wsadowych i ich przetworzenie do formatu GeoPackage

1. Cel etapu

Celem tego etapu jest przetworzenie testowych danych wsadowych, obejmujących dane geofizyczne, geologiczne i hydrogeologiczne, oraz konwersja tych danych do formatu

GeoPackage (GPKG). GeoPackage to otwarty format plików umożliwiający przechowywanie danych geoprzestrzennych, który jest wydajny i łatwy w użyciu.

2. Wymagania dotyczące danych

1. Źródła danych:

- **Geofizyczne:** Pliki z danymi sejsmicznymi, magnetycznymi, grawimetrycznymi i elektrooporowymi.
- **Geologiczne:** Dane o warstwach skalnych, ich właściwościach fizycznych i chemicznych, mapy geologiczne.
- **Hydrogeologiczne:** Informacje dotyczące poziomów wód gruntowych, przepuszczalności warstw, chemizmu wód podziemnych.

2. Formaty danych wejściowych:

- Pliki CSV, SHP (Shapefile), GeoTIFF, KML, GDB (Geodatabase), itp.

3. Narzędzia i oprogramowanie

- **GIS (Geographic Information System):** QGIS, ArcGIS
- **Oprogramowanie do analizy danych i konwersji formatów:** GDAL (Geospatial Data Abstraction Library), ogr2ogr
- **Języki programowania:** Python (biblioteki geopandas, fiona, shapely)

4. Kroki w geoprocessingu i konwersji do GeoPackage

1. Zbieranie i przygotowanie danych wsadowych:

- **Kompilacja danych:** Zgromadzenie wszystkich testowych danych wsadowych w różnych formatach.
- **Weryfikacja jakości:** Sprawdzenie kompletności i dokładności danych, usunięcie duplikatów, weryfikacja poprawności współrzędnych.

2. Normalizacja danych:

- **Przekształcenie układów współrzędnych:** Upewnienie się, że wszystkie dane są w jednolitym układzie współrzędnych (np. WGS 84).
- **Formatowanie atrybutów:** Standaryzacja nazw atrybutów i ich jednostek miar.

3. Geoprocessing:

- **Przestrzenne przetwarzanie danych:** Wykonanie operacji takich jak łączenie warstw, klipowanie, buforyzacja, przetwarzanie rastrów, itp.
- **Analizy przestrzenne:** Przeprowadzenie niezbędnych analiz geoprzestrzennych (np. interpolacja, analizy przestrzenne).

4. Konwersja do formatu GeoPackage:

- **Użycie narzędzi GDAL/ogr2ogr:** Wykorzystanie GDAL do konwersji różnych formatów plików do GeoPackage.

```
sh
przykładowy kod
ogr2ogr -f "GPKG" output.gpkg input.shp
```

- **Skryptowanie w Pythonie:** Automatyzacja procesu konwersji przy użyciu skryptów Python.

```
python
przykładowy kod
import geopandas as gpd
# Wczytanie danych z pliku SHP
data = gpd.read_file('input.shp')
# Zapisanie danych do formatu GeoPackage
data.to_file('output.gpkg', driver='GPKG')
```

5. Weryfikacja i walidacja danych GeoPackage:

- **Sprawdzenie integralności danych:** Upewnienie się, że wszystkie dane zostały poprawnie przetworzone i skonwertowane.
- **Testowanie w środowisku GIS:** Wczytanie plików GeoPackage do oprogramowania GIS (np. QGIS) w celu sprawdzenia poprawności i kompletności danych.

6. Dokumentacja procesu:

- **Sporządzenie dokumentacji:** Opis procesu geoprocessingu i konwersji, użytych narzędzi, skryptów oraz ewentualnych problemów napotkanych podczas przetwarzania.
- **Instrukcje użytkownika:** Przygotowanie krótkiego przewodnika dla użytkowników dotyczącego korzystania z danych GeoPackage.

5. Przykłady operacji geoprocessingu

1. Łączenie warstw:

- Łączenie różnych warstw geologicznych w jeden spójny zbiór danych.

2. Klipowanie:

- Wycinanie danych do określonego obszaru zainteresowania.

3. Buforyzacja:

- Tworzenie stref buforowych wokół punktów pomiarowych.

4. Interpolacja:

- Generowanie ciągłych powierzchni na podstawie punktowych danych.

6. Wnioski i dalsze kroki

Po zakończeniu etapu geoprocessingu i konwersji danych do formatu GeoPackage, należy zintegrować te dane z istniejącymi systemami zarządzania danymi oraz przetestować ich funkcjonalność w kontekście rzeczywistych zastosowań. Konieczne będzie również szkolenie zespołu w zakresie obsługi i analizy danych GeoPackage oraz monitorowanie ich efektywności i wprowadzanie ewentualnych usprawnień.

III. Przygotowanie podstaw modelu wizualizacji danych przestrzennych w serwisie internetowym.

1. Cel etapu

Celem tego etapu jest stworzenie podstawowego modelu wizualizacji danych przestrzennych, takich jak dane geofizyczne, geologiczne i hydrogeologiczne, w serwisie internetowym. Model ten powinien umożliwiać interaktywną przeglądanie, analizę i prezentację danych przestrzennych użytkownikom końcowym.

2. Wymagania dotyczące wizualizacji

1. Interaktywność:

- Możliwość przeglądania, przybliżania, oddalania oraz przesuwania map.
- Interaktywne narzędzia do pomiaru, rysowania i selekcji.

2. Czytelność i użyteczność:

- Przejrzyste prezentowanie danych z odpowiednimi etykietami i legendami.
- Intuicyjny interfejs użytkownika dostosowany do różnych poziomów zaawansowania użytkowników.

3. Integracja z bazami danych:

- Możliwość pobierania i aktualizacji danych z centralnej bazy danych.

3. Narzędzia i technologie

• Frontend (warstwa kliencka):

- HTML, CSS, JavaScript
- Biblioteki do wizualizacji map: Leaflet, OpenLayers, Mapbox GL JS
- Frameworki JavaScript: React, Vue.js, Angular

• Backend (warstwa serwerowa):

- Serwer aplikacji: Node.js, Flask, Django
- Baza danych: PostgreSQL z PostGIS, MySQL, MongoDB
- Serwery map: GeoServer, MapServer

• APIs i formaty danych:

- WMS (Web Map Service), WFS (Web Feature Service)
- GeoJSON, TopoJSON

4. Kroki w przygotowaniu modelu wizualizacji

1. Projektowanie architektury systemu:

- **Warstwa frontendowa:** Wybór biblioteki do wizualizacji map i frameworku JavaScript.
- **Warstwa backendowa:** Wybór serwera aplikacji i bazy danych.
- **Interfejsy API:** Określenie formatów danych i standardów komunikacji między frontendem a backendem.

2. Instalacja i konfiguracja środowiska:

- Instalacja narzędzi do tworzenia serwisu internetowego (np. Node.js, npm, framework wybranego języka).
- Konfiguracja serwera map (np. GeoServer) do obsługi danych przestrzennych.

3. Przygotowanie i import danych:

- Import danych przestrzennych do bazy danych (PostGIS, GeoServer).
- Przygotowanie warstw map, stylów i legend.

4. Tworzenie interfejsu użytkownika:

- **Podstawowa mapa:** Dodanie mapy bazowej (np. OpenStreetMap) i warstw danych przestrzennych.
- **Narzędzia interaktywne:** Implementacja narzędzi do przeglądania, przybliżania, oddalania i pomiarów.
- **Panele informacyjne:** Dodanie legend, etykiet i okien informacyjnych.

5. Integracja z backendem:

- Implementacja serwisów API do pobierania i aktualizacji danych przestrzennych.
- Obsługa zapytań do bazy danych (np. zapytania przestrzenne w PostGIS).

6. Testowanie i walidacja:

- Testowanie funkcjonalności serwisu na różnych urządzeniach i przeglądarkach.
- Walidacja poprawności wyświetlanych danych i interakcji użytkownika.

7. Dokumentacja i szkolenie:

- Przygotowanie dokumentacji technicznej opisującej architekturę systemu, konfigurację i sposób korzystania z serwisu.
- Przeprowadzenie szkoleń dla użytkowników końcowych i administratorów systemu.

5. Przykłady implementacji

1. Podstawowa mapa:

```
html
przykładowy kod
<!DOCTYPE html>
<html>
<head>
  <title>Mapa Przestrzenna</title>
  <link rel="stylesheet" href="https://unpkg.com/leaflet/dist/leaflet.css" />
  <style>
    #map {
      height: 100vh;
    }
  </style>
</head>
<body>
  <div id="map"></div>
  <script src="https://unpkg.com/leaflet/dist/leaflet.js"></script>
  <script>
    var map = L.map('map').setView([51.505, -0.09], 13);
    L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
      maxZoom: 19
    }).addTo(map);
  </script>
</body>
</html>
```

2. Dodanie warstw danych:

```
javascript
przykładowy kod
// Przygotowanie warstwy GeoJSON
var geojsonLayer = L.geoJson(geojsonData, {
  style: function (feature) {
    return {color: feature.properties.color};
  }
}).addTo(map);

// Przykład zapytania AJAX do pobrania danych z backendu
fetch('/api/dane-przestrzenne')
  .then(response => response.json())
  .then(data => {
    L.geoJson(data).addTo(map);
  });
```

3. Integracja z backendem:

```
python
przykładowy kod
from flask import Flask, jsonify
from flask_sqlalchemy import SQLAlchemy
```

```

app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'postgresql://user:password@localhost/dbname'
db = SQLAlchemy(app)

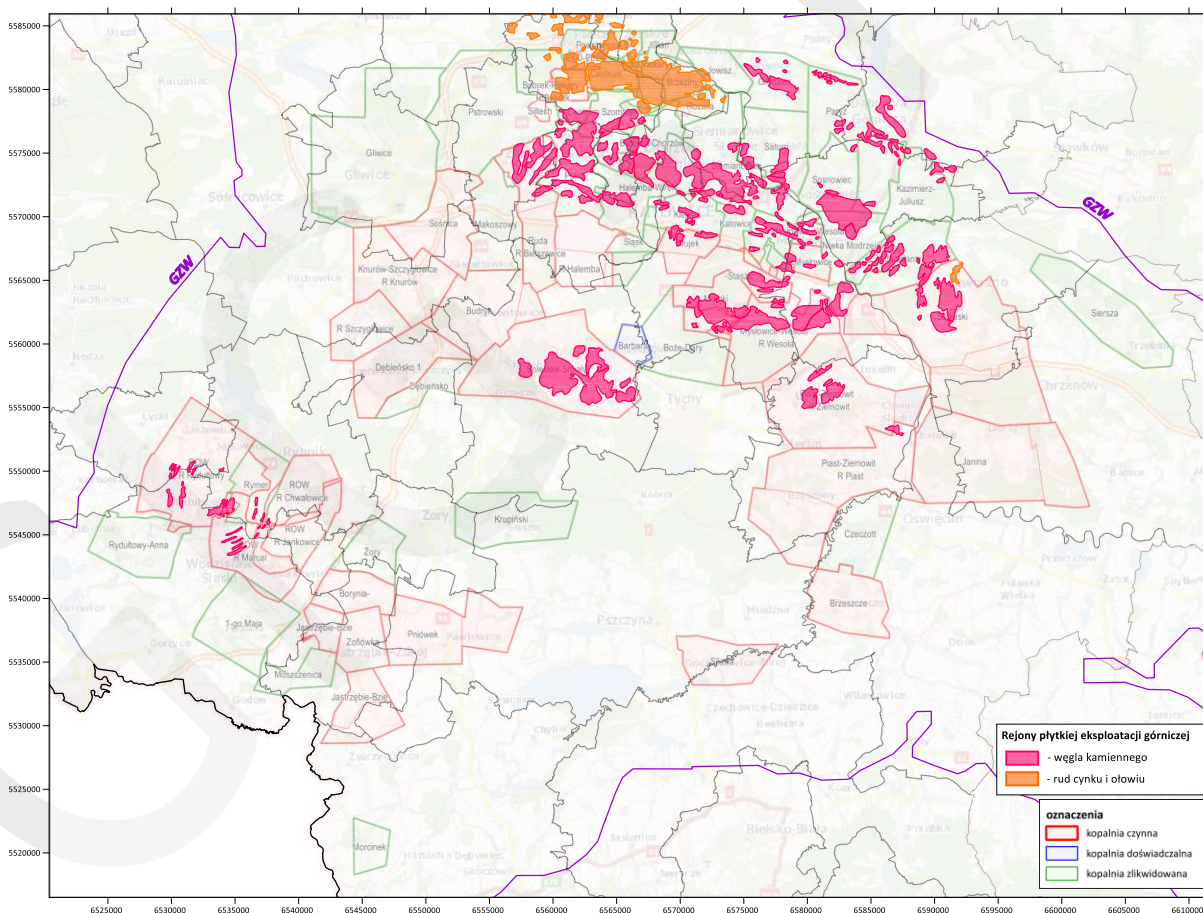
@app.route('/api/dane-przestrzenne')
def dane_przestrzenne():
    results = db.session.execute('SELECT * FROM dane_przestrzenne').fetchall()
    geojson_data = convert_to_geojson(results)
    return jsonify(geojson_data)

if __name__ == '__main__':
    app.run(debug=True)

```

6. Wnioski i dalsze kroki

Po stworzeniu podstawowego modelu wizualizacji danych przestrzennych w serwisie internetowym, kolejnym krokiem będzie dodanie zaawansowanych funkcjonalności, takich jak filtrowanie danych, analiza przestrzenna i eksport danych. Ważne jest również monitorowanie wydajności serwisu oraz zbieranie opinii użytkowników w celu wprowadzania ulepszeń i optymalizacji.



Rys. 1. Mapa GZW z przebiegiem granic obszarów górniczych czynnych i zlikwidowanych zakładów górniczych oraz rejonami płytkiej eksploatacji górniczej – przykład prezentacji danych na tle mapy podstawowej