

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ АНАЛІЗУ ДАНИХ МОНІТОРИНГУ СТАНУ ПОВЕРХНЕВИХ ВОД БАСЕЙНУ Р. ПІВДЕННИЙ БУГ

¹Вінницький національний технічний університет

Анотація

У роботі розглянуто інформаційну технологію аналізу даних моніторингу стану поверхневих вод басейну р. Південний Буг. Запропоновано методику парсингу, обробки та візуалізації даних, що містять вимірювання концентрацій забруднюючих речовин. Виконано автоматизоване зчитування даних із таблиць, їх структурування та інтеграцію з географічними координатами постів спостереження. Розроблено засоби візуалізації результатів на карті та у вигляді графіків змін показників у часі. Отримані результати можуть бути використані для оцінки екологічного стану водних ресурсів та прийняття рішень у сфері моніторингу довкілля.

Ключові слова: моніторинг води, парсинг даних, поверхневі води, Південний Буг, аналіз забруднень, геовізуалізація, якість води.

Abstract

The paper considers the information technology for analyzing data on monitoring the state of surface waters in the Southern Bug River basin. A methodology for parsing, processing, and visualizing data containing measurements of concentrations of harmful compounds is proposed. The automated reading of data from tables, their structuring and integration with the geographical coordinates of observation posts were performed. Tools for visualizing the results on a map and in the form of graphs of changes in parameters over time were developed. The obtained results can be used to assess the ecological state of water resources and make decisions in the field of environmental monitoring.

Keywords: water monitoring, data parsing, surface waters, Southern Bug River, pollution analysis, geovisualization, water quality

Вступ

У цій роботі розглядається розробка інформаційної технології для аналізу даних моніторингу стану поверхневих вод басейну річки Південний Буг. Основна увага приділяється методам обробки даних, автоматизованому збору інформації та візуалізації отриманих результатів у вигляді картографічних матеріалів.

Моніторинг якості поверхневих вод має критичне значення для оцінки екологічного стану водних об'єктів і розробки заходів щодо їх охорони. Зокрема, басейн річки Південний Буг є важливим джерелом водопостачання для багатьох регіонів України, тому моніторинг його стану є актуальним завданням.

Метою даного дослідження є розробка інформаційної технології для автоматизованого аналізу та візуалізації даних моніторингу стану поверхневих вод басейну річки Південний Буг.

Створення набору даних

Візуальне структурне представлення запропонованого підходу, створене за допомогою сервісу diagrams.net [1], приведене на (рис. 1).

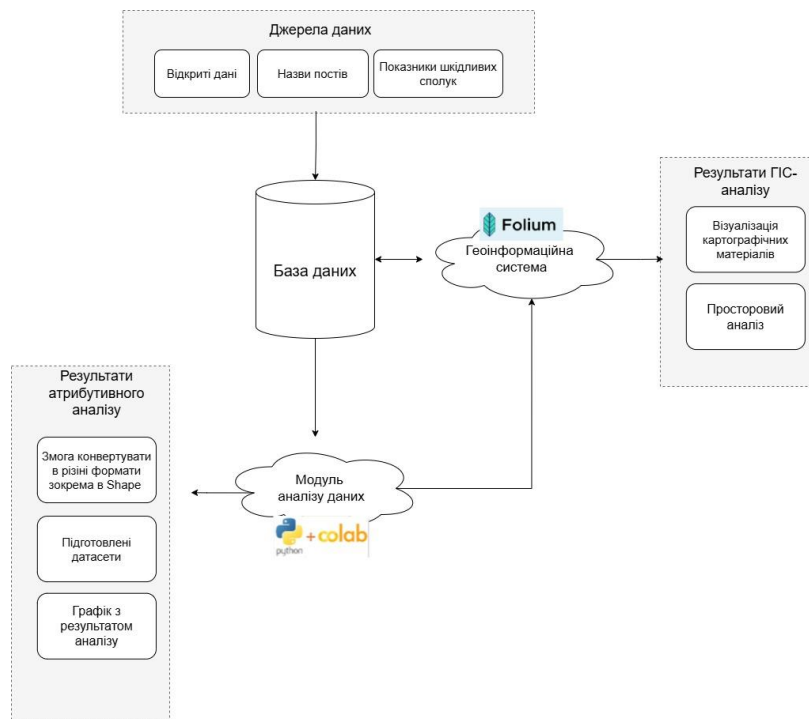


Рисунок 1. Візуальне структурне представлення запропонованого підходу

Використано дані моніторингу якості води від офіційного ресурсу EcoWaterMon [2], який містить відомості про концентрацію різних забруднюючих речовин у поверхневих водах. Це дозволило отримати актуальні показники стану річки Південний Буг та провести їх подальший аналіз (рис. 2).

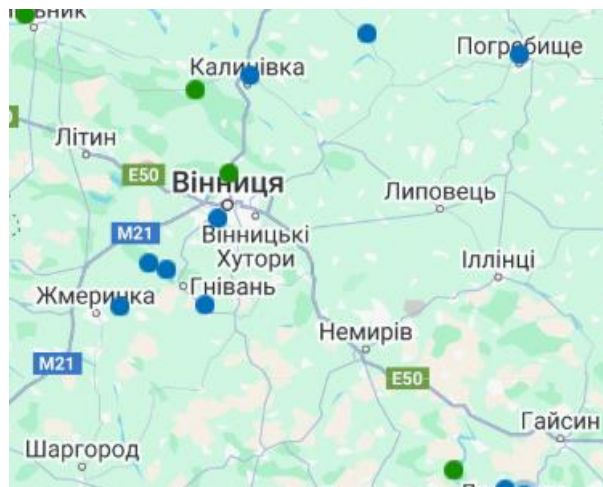


Рисунок 2. Станції системи моніторингу якості води річки Південний Буг неподалік міста Вінниці

Після отримання даних моніторингу поверхневих вод, необхідно забезпечити їх підготовку для подальшого аналізу, що включає парсинг вихідних файлів (рис. 3). Парсинг — це процес автоматичного витягування потрібної інформації з наданих файлів. В даному випадку, необхідно отримати такі дані, як назви постів спостереження, дати вимірювань та концентрації різних речовин, що вимірюються на кожному з постів.

Вихідні файли можуть бути представлені в різних форматах, таких як CSV (Comma-Separated Values) або Excel (XLS, XLSX). Для кожного формату парсинг передбачає використання відповідних інструментів або бібліотек для витягування даних. Наприклад, для роботи з CSV можна використати бібліотеки Python, такі як csv або pandas, а для файлів Excel — бібліотеку openpyxl або pandas.

Google Colab [3] дозволяє працювати з бібліотеками pandas, numpy та matplotlib, що забезпечує ефективну обробку та аналіз даних.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Дані моніторингу (за районом річкового басейну або суббасейну)											
2	з 01.03.2000 до 11.03.2025											
3	Район											
4	1. Пост: Балка Ташлик, 6 км, Ташлицьке вдсх., став-охолоджувач ПУ АЕС (міст через Ташлицьке вдсх., південь м. Південно-Українськ)											
5	Значення	Показник										
6		Амоній-іони, мг	Біохімічне спож	Завислі (с	Кисень рс	Нітрат-іон	Нітрит-іон	Сульфат-іс	Фосфат-іс	Хлорид-іони, мг/дм3		
7	Дата											
8	17.04.2000	0,035	1,3	0	12,98	4,7	0,038	403,2	0,143	146,1		
9	04.07.2000	0,031	1,13	10,4	9,89	0,24	0,022	416	0,06	149		
10	04.12.2000	0,042	1,2	0	9,63	2,02	0,006	384	0,275	154,5		
11	06.02.2001	0,062	2,88	1,6	11,77	4,3	0,004	364,8	0,38	160,1		
12	03.04.2001	0	2,5	15,6	11,4	11,4	0,023	397	0,197	146		
13	09.07.2001	0,05	1,98	21	12,97	0,51	0,022	384	0,002	154,5		
14	08.10.2001	0,106	1,04	0	9,63	0,86	0,007	364,8	0,061	148,9		
15	09.01.2002	0	1,77	12	12,04	3,78	0,032	317,2	0,315	168,4		

Рисунок 3. Початковий вигляд отриманих даних з ресурсу

Ця частина коду призначена для парсингу даних про якість води (рис. 4). Спочатку вхідний текст розбивається на рядки, а потім витягується інформація про пости спостереження (номер і назва), показники якості води та вимірювання. Для цього використовуються регулярні вирази, щоб знайти відповідні рядки.

Код перевіряє наявність заголовків для показників, після чого розбиває дані на окремі значення, перетворюючи їх на числа. Якщо це неможливо, значення вважається відсутнім. Наприкінці дані про пости та вимірювання зберігаються у двох фреймах для подальшого аналізу.

```

for line in lines:
    line = line.replace('""', '').strip(' ')
    post_match = re.match(r'(\d+)\.\\s*Пост:\s*(.*)?;$', line)
    if post_match:
        post_number = int(post_match.group(1))
        post_name = post_match.group(2).strip() # Прибираємо зайві пробіли
        current_post = {'post_number': post_number, 'post_name': post_name}
        monitoring_posts.append(current_post)
        in_data_block = False
        headers_found = False
        continue
    if not headers_found and 'Показник' in line:
        headers_found = True
        continue
    if headers_found and not in_data_block and len(line) > 0 and ';' in line:
        indicators = [ind.strip() for ind in line.rstrip(';').split(';')]
        in_data_block = True
        continue
    if in_data_block and 'Дата' in line:
        continue
    if in_data_block and len(line) > 0 and ';' in line:
        values = line.rstrip(';').split(';')
        if len(values) > 0 and values[0].strip() and re.match(r'\d{2}\.\d{2}\.\d{4}', values[0]):
            date = values[0].strip()
            row_data = {'post_number': post_number, 'date': date}
            for i, value in enumerate(values[1:], 1):
                if i <= len(indicators) and value.strip():
                    try:
                        row_data[indicators[i-1]] = float(value.replace(',', '.'))
                    except ValueError:
                        row_data[indicators[i-1]] = None
            measurements.append(row_data)

```

Рисунок 4. Фрагмент коду для парсингу даних

Отримані після парсингу файли містять структуровану інформацію для подальшого аналізу та були вручну підкориговані, зокрема додано два нові стовпці — lat і lon для відображення постів на карті (рис. 5). Координати постів визначалися за допомогою Google Maps [4]. Файл name_coord_posts.csv містить назви постів спостереження та їх унікальні номери (post_number), що дозволяє ідентифікувати кожен пост і прив'язати до нього відповідні вимірювання.

```
df_post = pd.read_csv('name_coord_posts.csv')
df_post
```

	post_number	post_name	lat	lon
0	1	Балка Ташлик, 6 км, Ташлицьке вдсх., став-охол...	49.521558	28.848859
1	2	Бузький лиман, 0,5 км, скид МКП "Миколаїводока...	49.508086	26.759621
2	3	Возсіятське водосховище, 0 км, у межах села В...	49.446399	28.529845
3	4	р. Баран, 17 км, м. Жмеринка	49.439776	26.964037
4	5	р. Велика Вись, 95 км, с. Лікареве, Новомиргор...	49.434142	27.412395
5	6	р. Гнилий Тікич, 41 км, м. Звенигородка, питни...	49.422267	28.380992
6	7	р. Грузька, 4 км, Лелеківське вдсх., Кіровогра...	49.412291	27.630160
7	8	р. Десна, 0 км, с. Самгородок Козятинський р-н...	49.393093	27.119108
8	9	р. Інгул, 0 км, у межах села Привільне, міст н...	49.389246	28.383234

Рисунок 5. Датасет з назвами постів та їх координатами

Другий файл, `measurement`, містить результати моніторингу якості води (рис. 6). У ньому записані значення концентрацій різних показників (наприклад, амоній, розчинений кисень, нітрати, фосфати тощо) для кожного поста (`ID_Station`) у певну дату (`Date`). Відсутні значення позначені як `NaN`, що враховується під час подальшої обробки.

```
df_data = pd.read_csv('measurement.csv')
df_data
```

	ID_Station	Date	NH4	BSK5	Suspended	O2	NO3	NO2	SO4	PO4	Cl
0	1	4/17/2000	0.035	1.30	NaN	12.98	4.70	0.038	403.2	0.143	146.1
1	1	7/4/2000	0.031	1.13	10.4	9.89	0.24	0.022	416.0	0.060	149.0
2	1	12/4/2000	0.042	1.20	NaN	9.63	2.02	0.006	384.0	0.275	154.5
3	1	2/6/2001	0.062	2.88	1.6	11.77	4.30	0.004	364.8	0.380	160.1
4	1	4/3/2001	NaN	2.50	15.6	11.40	11.40	0.023	397.0	0.197	146.0

Рисунок 6. Датасет з результатами моніторингу в постах

Результати дослідження

Для створення карти використовується бібліотека `folium`, яка дозволяє візуалізувати географічні дані на інтерактивній мапі. Додаючи назви постів на маркери, ми використовуємо інформацію з файлу `name_coord_posts.csv`, де зберігаються координати (`lat`, `lon`) та назви постів (`post_name`). Для кожного запису створюється маркер на карті `folium`, в якому зазначається розташування поста (рис. 7).

```
# Створення карти
m = folium.Map(location=[48.688391, 30.347386], zoom_start=8)

# Додаємо маркер на карту
marker = folium.Marker(
    location=[float(df_post.loc[i, 'lat']), float(df_post.loc[i, 'lon'])],
    popup=folium.Popup(popup_text),
    icon=folium.Icon(icon="industry", prefix="fa", color="blue")
)
marker.add_to(m)
```

Рисунок 7. Фрагмент коду для додавання маркерів з назвами на карту

У рорир передається не лише назва поста, а й побудований графік у вигляді зображення, закодованого у форматі Base64 (рис. 8). Це дозволяє інтерактивно переглядати зміни показників якості води безпосередньо на карті.

```
# Створюємо графік
plt.figure(figsize=(15, 10))
for column in ["NH4", "BSK5", "Suspended", "O2", "NO3", "NO2", "SO4", "PO4", "Cl"]:
    plt.plot(pd.to_datetime(df_measurements_post["Date"]), df_measurements_post[column], label=column)

plt.xlabel("Дата")
plt.ylabel("Концентрація")
plt.title(f"Виміри для посту {post_id}")
plt.legend()
plt.grid(True)
plt.xticks(rotation=45)

# Зберігаємо графік у пам'яті
img = BytesIO()
plt.savefig(img, format='png', bbox_inches="tight")
img.seek(0)
plt.close()
# Конвертуємо зображення у base64
img_base64 = base64.b64encode(img.getvalue()).decode()
```

Рисунок 8. Фрагмент коду побудови графіку

Графік зберігається у форматі png, конвертується у Base64 (img_base64), після чого вставляється в рорир за допомогою HTML-елемента . Це дозволяє зручно аналізувати дані, оскільки при натисканні на маркер користувач одразу бачить як місце поста, так і візуалізацію результатів вимірювань в ньому.

У результаті, при натисканні на маркер річки відкривається спливаюче вікно, у якому відображається назва поста спостереження та графік. Графік показує динаміку змін концентрацій шкідливих речовин у воді протягом часу, що дозволяє швидко оцінити тенденції забруднення в конкретному місці (рис. 9).

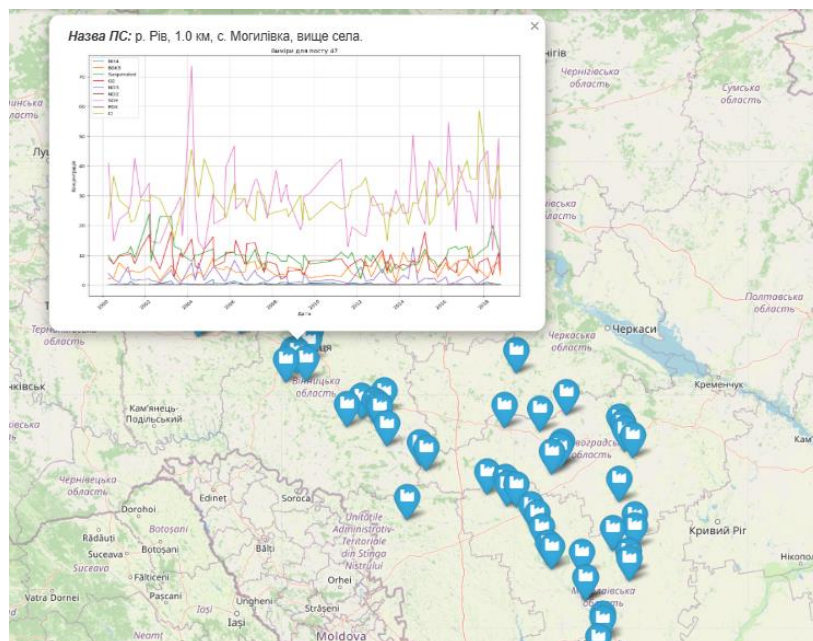


Рисунок 9. Інтерактивна карта моніторингу вод

Так, карту, створену за допомогою бібліотеки folium, можна зберегти в форматі HTML за допомогою методу m.save("map.html"). Це дозволяє зберегти інтерактивну карту, яку потім можна додати на веб-сайт або вбудувати в інші веб-додатки.

Збережений файл буде містити всю структуру карти, маркери, спливаючі вікна, графіки та інші елементи, що дозволяє користувачам взаємодіяти з картою без необхідності встановлення додаткових бібліотек.

Для збереження географічних даних у форматі Shapefile, використовуємо бібліотеку geopandas. Спочатку створюється GeoDataFrame, який містить координати широти та довготи для кожного поста спостереження. Для цього генерується геометрія у вигляді точок, використовуючи координати з файлу df_post. Для кожного поста формується точка на основі його координат, що дозволяє вказати точне місце розташування на карті (рис. 10).

```
# Створюємо GeoDataFrame
geometry = [Point(lon, lat) for lat, lon in zip(df_post["lat"], df_post["lon"])]
gdf = gpd.GeoDataFrame(df_post, geometry=geometry)

# Збереження у форматі Shapefile
gdf.to_file("map.shp", driver="ESRI Shapefile")
```

Рисунок 10. Збереження карти у Shapefile

Після цього, створений GeoDataFrame зберігається у форматі Shapefile за допомогою методу to_file(). Цей файл, збережений у форматі map.shp, можна використовувати в ГІС-застосунках, таких як QGIS або ArcGIS для подальшого аналізу чи візуалізації даних (рис. 11). Це дозволяє інтегрувати дані з інших джерел або доповнювати карту іншими шарами в спеціалізованих програмах для роботи з географічною інформацією [5-7].

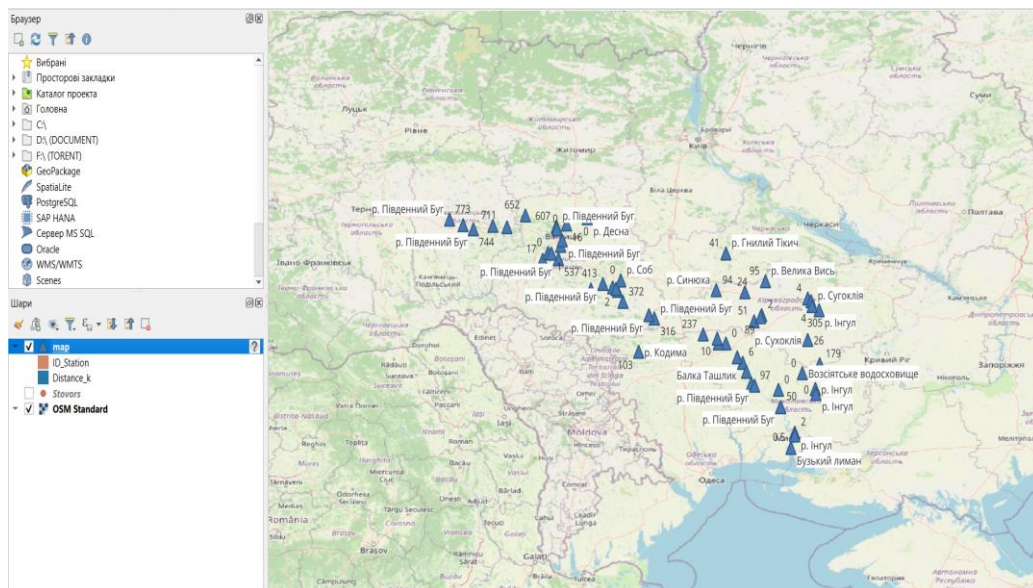


Рисунок 11. Шар з постами, відкритий в ГІС QGIS

Висновки

У цій роботі було розглянуто розробку інформаційної технології для автоматизованого аналізу та візуалізації даних моніторингу стану поверхневих вод басейну річки Південний Буг. Запропонований підхід дозволяє ефективно обробляти великі обсяги екологічних даних, автоматизувати збір та аналіз інформації, а також створювати інтерактивні карти для візуалізації результатів. Це сприяє покращенню оцінки екологічного стану водних ресурсів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Безкоштовний онлайн-інструмент для створення діаграм та схем [Електронний ресурс] – Режим доступу до ресурсу: <https://app.diagrams.net/>.
2. Моніторинг стану водойм [Електронний ресурс] – Режим доступу до ресурсу: <http://monitoring.davr.gov.ua/EcoWaterMon/GDKMap/Index>.
3. Безкоштовний хмарний сервіс для виконання Python-коду: [Електронний ресурс]. – URL: <https://colab.research.google.com/>.
4. Безкоштовний хмарний сервіс Google Maps: [Електронний ресурс]. – URL: <https://www.google.com/maps/>.
5. Наука про дані: машинне навчання та інтелектуальний аналіз даних : електронний навчальний посібник комбінованого (локального та мережевого) використання [Електронний ресурс] / В. Б. Мокін, М. В. Дратовий – Вінниця : ВНТУ, 2024. – 258 с.
5. Мокін В. Б. Методи оцінювання та засоби автоматизації розрахунку складових водогосподарського балансу районів річкових басейнів України [Електронний ресурс] : монографія / В. Б. Мокін, В. В. Гребінь, Є. М. Крижановський. – Електрон. текст. дані (PDF: 5,4 Мб). – Вінниця : ВНТУ, 2023. – 168 с.

6. Метод аугментації текстів про стан масивів вод на основі інтелектуальної прив'язки до багатозв'язних геоінформаційних систем іменованих сутностей [Текст] / В. Б. Мокін, К. О. Бондалетов, Є. М. Крижановський, і В. О. Караваєв // Вісник Вінницького політехнічного інституту. – 2023. – Вип. 3. – С. 55–65.

Євгеній Миколайович Крижановський – канд. техн. наук, доцент кафедри системного аналізу та інформаційних технологій, Вінницький національний технічний університет, Вінниця, e-mail: kruzhan@gmail.com;

Караваєв Вадим Олександрович – студент групи 2ІСТ-24м, Факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: karavaevvadim1999@gmail.com;

Ігор Миколайович Штельмах – канд. техн. наук, асистент кафедри системного аналізу та інформаційних технологій, Вінницький національний технічний університет, Вінниця, e-mail: igor.shtelmakh@vntu.edu.ua.

Evgeniy Kryzhanovsky M. – Cand. Sc. (Eng), Department of Systems Analysis and Information Technology, Vinnytsia National Technical University, Vinnytsia, e-mail: kruzhan@gmail.com;

Karavaev Vadim O. – student of group 2IST-24m, Faculty of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: karavaevvadim1999@gmail.com;

Shtelmah, Igor M. – candidate of technical sciences, assistant professor of the Department of System Analysis and Information Technologies, Vinnytsia National Technical University, Vinnytsia, e-mail: igor.shtelmakh@vntu.edu.ua.