

**КРИЛИК ЛЮДМИЛА**

Вінницький національний технічний університет

<https://orcid.org/0000-0001-6642-754X>

e-mail: lyudmila.krylik@gmail.com

**ШАРГАЛО ДМИТРО**

Вінницький національний технічний університет

e-mail: bigdcoolboy@gmail.com

## РОЗШИРЕННЯ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ ПРОГРАМНОГО МОДУЛЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ УПРАВЛІННЯ ВІДНОСИНАМИ З КЛІЄНТАМИ

Робота присвячена розширенню функціональних можливостей програмного модуля інформаційної технології управління відносинами з клієнтами (CRM). Для реалізації розробки була обрана клієнт-серверна архітектура. Клієнтська частина реалізована за допомогою JavaScript та фреймворку React. Для розробки серверної частини використано Node.js. Для управління базою даних обрано MongoDB. Середовищем розробки слугував Visual Studio Code. Для реалізації розробки, забезпечення швидкої роботи системи та подальшої її підтримки застосовано модульний підхід. З метою кращого розуміння роботи програмного модуля CRM розроблена UML-діаграма послідовності. Розробка успішно пройшла тестування. Відповідає всім заданим вимогам та має розширений функціонал у порівнянні з аналогами щонайменше за 4 можливостями: наявність інтегрованого чату для швидкої комунікації з клієнтами; можливість додавання нотаток, які на відміну від аналогів мають спрощений механізм додавання нотаток; оптимізований інтерфейс, що робить роботу з програмою швидшою та інтуїтивно зрозумілою; конкурентоспроможна ціна, що є привабливим для малого та середнього бізнесу. Розширений функціонал сприятиме підвищенню ефективності управління відносинами з клієнтами. Розроблений програмний продукт на відміну від аналогів займає менший обсяг пам'яті на сервері.

Ключові слова: управління відносинами з клієнтами, CRM, інформаційна технологія, автоматизація, персоналізація, аналіз даних.

**KRYLIK LYUDMILA****SHARGALO DMYTRO**

Vinnytsia National Technical University

## EXPANDING THE FUNCTIONAL CAPABILITIES OF THE INFORMATION TECHNOLOGY SOFTWARE MODULE FOR CLIENT RELATIONSHIP MANAGEMENT

The work is dedicated to expanding the functional capabilities of the Customer Relationship Management (CRM) software module. A client-server architecture was chosen for the implementation. The client side is developed using JavaScript and the React framework. Using React allows for the creation of a convenient and intuitive interface for client interaction. For the server side, Node.js was used. MongoDB, a document-oriented NoSQL database, was chosen for database management. This type of database is ideal for handling large volumes of data and offers flexibility in data storage structure. Visual Studio Code was selected as the development environment, which provides numerous advantages, including open-source code, ease of use, intelligent autocompletion, an integrated debugger, Git integration, and support for various extensions that significantly enhance development productivity. A modular approach was used in the development to ensure fast system performance and ease of future maintenance. To better understand the operation of the CRM software module, a UML sequence diagram was developed, which is an important part of the software development process. The development was successfully tested. It meets all the specified requirements and has extended functionality compared to similar systems, offering at least four key advantages: the presence of an integrated chat for fast communication with clients; the ability to add notes, which, unlike analogs, feature a simplified note-taking mechanism; an optimized interface, making the program faster and more intuitive to use; and a competitive price, making it attractive for small and medium-sized businesses. This extended functionality will contribute to improving customer relationship management efficiency. The developed software product, unlike its counterparts, occupies less server memory.

Keywords: customer relationship management, CRM, information technology, automation, personalization, data analysis.

### Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями

Нині у сучасному, динамічному та висококонкурентному бізнес-середовищі, де технології стрімко розвиваються та очікування клієнтів зростають, компанії зіштовхуються з необхідністю не лише пропонувати високоякісні продукти та послуги, але й забезпечувати високий рівень обслуговування клієнтів. Основну перевагу у вирішенні цих питань мають CRM-системи, які сприяють підприємствам підтримувати та розвивати взаємовигідні відносини зі своїми клієнтами [1–4].

Наразі традиційні CRM-системи недостатньо гнучкі та адаптовані до специфічних потреб сучасного бізнесу. Це створює потребу в інноваціях та розробці нових технологічних рішень, які сприятимуть розробці та впровадженню в практичну діяльність підприємств простого, безкоштовного та багатофункціонального інструментарію управління взаємовідносинами з клієнтами [5–8].

Отже, розширення функціональних можливостей програмного модуля інформаційної технології управління відносинами з клієнтами є доцільним та має практичне значення.

### Аналіз останніх досліджень та публікацій

Успішна діяльність компанії в сучасному бізнес-середовищі залежить від ефективної роботи CRM-систем, а інноваційні CRM-рішення суттєво підвищують задоволеність та інтерес клієнтів до

нього. CRM-системи є незамінним інструментарієм сучасного бізнесу, який дозволяє будувати довгострокові відносини з клієнтами та надавати їм якісний сервіс. Вони сприяють підвищенню задоволеності клієнтів та ефективності протікання бізнес-процесів, надають можливість компаніям бути конкурентоспроможними.

Однак, існуючі рішення часто є комерційними сервісами, які не розкривають усіх деталей реалізації або мають обмежений функціонал. Це не дозволяє користувачам отримувати повноцінну інформацію для прийняття ґрунтовних рішень [1– 8].

#### Формулювання цілей статті

Метою роботи є розширення функціональних можливостей програмного модуля інформаційної технології управління відносинами з клієнтами за рахунок впровадження нового програмного інструментарію.

Для досягнення цієї мети потрібно вирішити такі задачі:

- 1) провести аналіз технічного рівня існуючих систем управління відносинами з клієнтами;
- 2) спроектувати архітектуру та структуру програмного модуля управління відносинами з клієнтами;
- 3) виконати програмну реалізацію програмного модуля інформаційної технології управління відносинами з клієнтами, спрямовану на розширення функціональних можливостей;
- 4) провести тестування розробленого програмного продукту;
- 5) зробити висновки з проведених досліджень.

#### Виклад основного матеріалу

Для реалізації програмного забезпечення програмного модуля інформаційної технології управління відносинами з клієнтами була розроблена його архітектура. Інформаційна технологія управління відносинами з клієнтами має клієнт-серверну архітектуру. Клієнтська частина реалізована за допомогою JavaScript та фреймворку React. Використання React дозволяє створити зручний та інтуїтивно зрозумілий інтерфейс для взаємодії з клієнтами [9–11]. Для розробки серверної частини використано Node.js [12]. Для управління базою даних обрано MongoDB. Це документоорієнтована NoSQL база даних, яка ідеально підходить для роботи з великими обсягами даних та забезпечує гнучкість у структурі збереження даних [13]. Середовищем розробки обрано Visual Studio Code, який надає численні переваги, зокрема відкритий вихідний код, зручність у використанні, інтелектуальне автозавершення, вбудований відлагоджувач, інтеграцію з Git та підтримку різноманітних розширень, що значно підвищує продуктивність розробки.

Зважаючи на те, що CRM-система має клієнт-серверну архітектуру, для реалізації розробки, забезпечення швидкої роботи системи та подальшої її підтримки застосовано модульний підхід. На рис. 1 подана загальна структура програмного модуля для управління відносинами з клієнтами.



Рис. 1. Загальна структура програмного модуля для управління відносинами з клієнтами

Крім того, для кращого розуміння роботи програмного модуля інформаційної технології управління відносинами з клієнтами розроблена UML-діаграма послідовності, яка є важливою частиною процесу розробки програмного забезпечення. Ця діаграма демонструє порядок обміну повідомленнями між об'єктами в системі в часовій послідовності та надає ґрунтовне уявлення про роботу програмного продукту, демонструючи взаємодію між різними елементами CRM-системи [14].

Розроблена UML-діаграма послідовності може містити такі основні компоненти, як клієнт, менеджер, CRM-система та база даних. На рис. 2 подано UML-діаграму послідовності взаємодії модулів інформаційної технології управління відносинами з клієнтами.

Розглянемо основні програмні рішення, які сприяють розширенню функціональних можливостей програмного модуля інформаційної технології управління відносинами з клієнтами. UML-діаграма послідовності (рис. 2) розробки містить чат, який слугує для швидкої комунікації з клієнтами. Програмна реалізація якого має такий зміст (рис. 3):

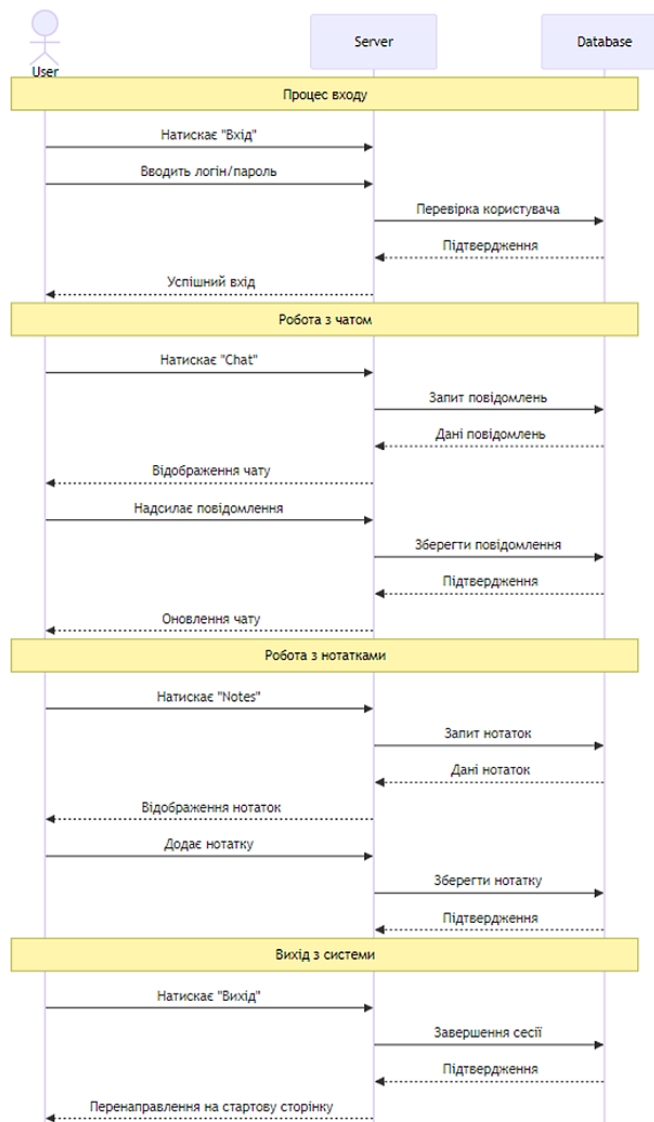


Рис. 2. UML-діаграма послідовності взаємодії модулів інформаційної технології управління відносинами з клієнтами

```

const [chats, setChats] = useState([]);
const [message, setMessage] = useState('');
const [chatId, setChatId] = useState('');
const [chatMessages, setChatMessages] = useState(null)
const { userData } = useSelector(state => state.auth)
const getChats = async () => {
  const chatsObj = await axios.get('/chat')
  setChats(chatsObj.data)
}
const subscribeGetMessages = async (id) => {
  try {
    console.log(id)
    setChatId(id)
    const messages = await axios.get(/message/${id});
    setChatMessages(messages)
    subscribeGetMessages(id)
  }
  catch (err) {
    setTimeout(() => {
      subscribeGetMessages(id)
    }, 500)
  }
}
React.useEffect(() => {
  getChats();
}, [])
const sendMessage = async () => {
  await axios.post('/message', { message, chatId
  setChatMessages([...chatMessages, {
    sender: userData.customer,
    message: message
  }])
}
const getMessages = async (id) => {
  const messages = await axios.get(/messages/${i
  setChatMessages(messages.data)
}
    
```

Рис. 3. Фрагмент програмної реалізації інтерфейсного вікна «Chat»

Функція `sendMessage` є асинхронною та використовується для надсилання нового повідомлення. Ця функція виконує POST-запит до API на ендпоінт `/message`, передає текст повідомлення та ідентифікатор чату. По завершенні надсилання повідомлення функція очищує поле введення, оновлює стан `message` та додає нове повідомлення до стану `chatMessages`, при цьому використовуються дані про користувача для зазначення відправника.

Функція `getMessages` є асинхронною та приймає `id` як параметр. Така функція виконує GET-запит до API на ендпоінт `/messages/${id}` для отримання повідомлень, які сполучені з певним чатом. Отримані повідомлення зберігаються у стані `chatMessages`.

На рис. 4. подано загальний вигляд інтерфейсного вікна «Chat», де візуалізується процес спілкування з клієнтом.

Chat

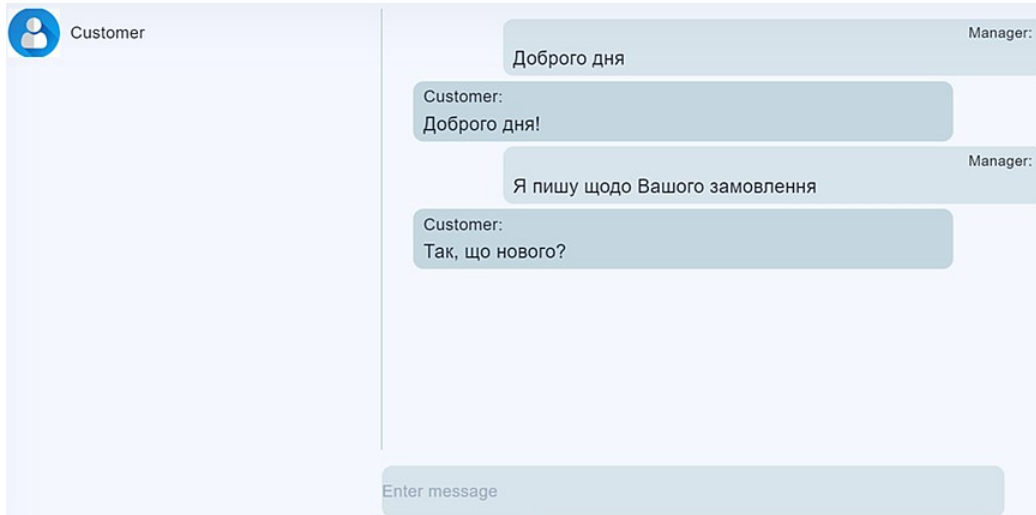


Рис. 4. Загальний вигляд інтерфейсного вікна «Chat»

Розробка успішно пройшла тестування та відповідає всім заданим вимогам. В табл. 1. подано результати тестування розробленого програмного продукту та його аналогів, а саме: Salesforce, HubSpot, Zoho CRM та Pipedrive.

Таблиця 1

Результати тестування розробленого програмного продукту та аналогів CRM-систем

Параметр	Розробка	Salesforce	HubSpot	Zoho CRM	Pipedrive
Інтегрований чат	+	-	-	+	-
Нотатки	+	+	+	+	+
Оптимізація інтерфейсу	+	+	+	-	-
Вартість	Низька	Висока	Середня	Середня	Середня

З табл. 1 видно, що розробка має розширений функціонал у порівнянні з аналогами щонайменше за 4 ключовими можливостями: інтегрований чат для швидкої комунікації з клієнтами; можливість додавання та зберігання нотаток, які на відміну від аналогів мають спрощений механізм додавання нотаток; оптимізація користувацького інтерфейсу, що робить роботу з програмою швидшою та інтуїтивно зрозумілою; доступна ціна, що робить систему привабливою для малого та середнього бізнесу.

Розширений функціонал сприятиме підвищенню ефективності управління відносинами з клієнтами. Крім того потрібно зауважити, що розроблений програмний продукт на відміну від аналогів займає менший обсяг пам'яті на сервері.

На рис. 5 для кращого сприйняття результатів тестування подано оцінювання функціональних можливостей розробки та аналогів за чотири бальною шкалою.

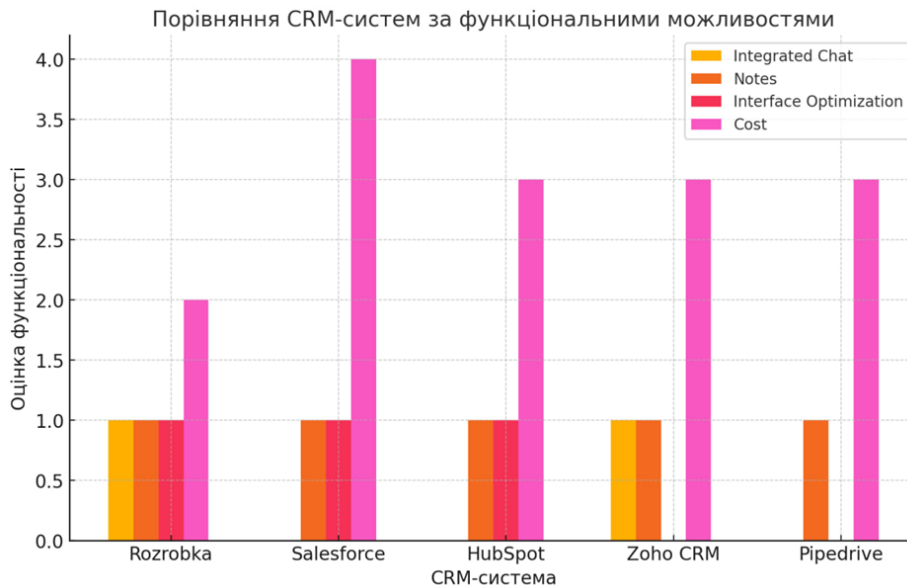
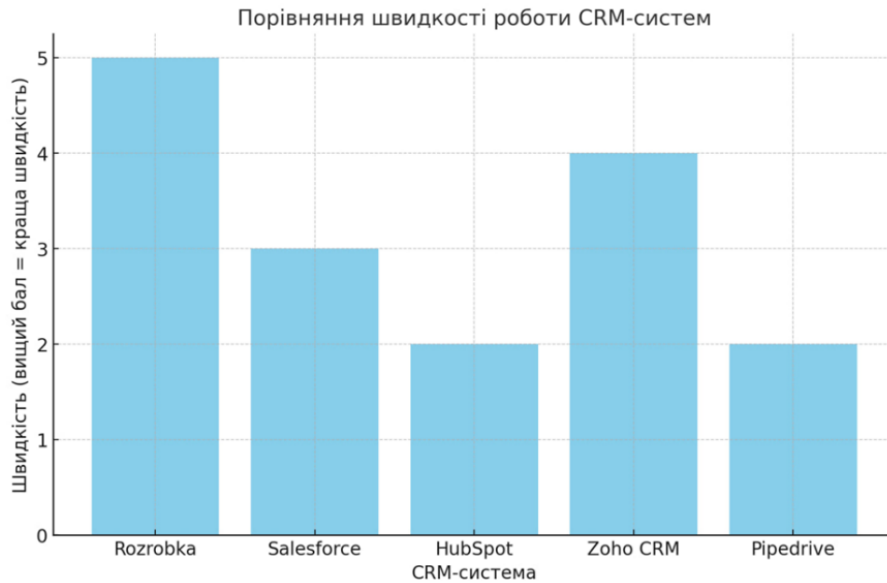


Рис. 5. Порівняння CRM-систем за їхніми функціональними можливостями

З рис. 5 видно, що інтегрований чат є тільки в Rozrobka і Zoho CRM (1 бал), тоді як у всіх інших системах цієї функції немає – 0 балів. Всі системи мають підтримку нотаток за однаковою оцінкою для цього параметра – 1 бал. Оптимізації інтерфейсу в Zoho CRM і Pipedrive відсутня, тобто мають низьку оцінку – 0 балів. Вартість є ще одним важливим параметром, яка оцінена як «Низька – 2 бали» для Rozrobka і «Висока – 4 бали» для Salesforce, інші системи мають середню вартість – 3 бали.

На рис. 6 подано порівняльні результати швидкості роботи розробки та аналогів: Salesforce, HubSpot, Zoho CRM та Pipedrive за п'яти бальною шкалою.



**Рис. 6. Порівняльні результати швидкості роботи розробленого програмного продукту та аналогів**

З рис. 6 видно, що найвищу швидкість роботи серед інших CRM-систем має Rozrobka – 5 балів, найменшу швидкість роботи має HubSpot та Pipedrive – 2 бали.

Отже, на основі результатів тестування можна зробити висновок, що поставлених цілей було досягнуто.

#### **Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі**

1. На основі аналізу існуючих технологічних рішень в галузі управління відносинами з клієнтами встановлено, що існуючі системи-аналоги, такі як CRM-платформи надають різноманітний програмний інструментарій для аналізу клієнтських даних та підтримки комунікації з ними. Проте, існуючі технологічні рішення найчастіше є комерційними сервісами, які не розкривають усіх деталей реалізації або мають дещо обмежений функціонал. Ця обставина не дозволяє користувачам отримувати повноцінну інформацію для прийняття потрібних рішень і слугувала поштовхом для подальших досліджень з метою розширення функціональних можливостей програмного модуля інформаційної технології управління відносинами з клієнтами.

2. Для розробки програмного модуля CRM обрано клієнт-серверну архітектуру та модульний підхід. Клієнтська частина реалізована за допомогою JavaScript та фреймворку React. Для розробки серверної частини використано Node.js. Управління базою даних здійснюється за допомогою MongoDB. Середовищем розробки слугує Visual Studio Code. Для кращого розуміння роботи програмного модуля CRM розроблена UML-діаграма послідовності.

3. Розробка успішно пройшла тестування та відповідає всім заданим вимогам. Програмний модуль інформаційної технології управління відносинами з клієнтами має розширений функціонал у порівнянні з системами-аналогами на 4 можливості: наявність інтегрованого чату для швидкої комунікації з клієнтами; можливість додавання нотаток, які на відміну від аналогів мають спрощений механізм додавання нотаток; оптимізований інтерфейс, що робить роботу з програмою швидшою та інтуїтивно зрозумілою; доступна ціна. Розробка в порівнянні з аналогами займає менший обсяг пам'яті на сервері.

#### **Література**

1. Шаргало Д. В Крилик Л. В. Перспективи розробки інформаційної технології управління відносинами з клієнтами. Матеріали ЛІІ науково-технічної конференції підрозділів Вінницького національного технічного університету, Вінниця, 20 – 22 березня 2024 р. URL: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2024/paper/view/20386/16891> (дата звернення: 27.11.2024).

2. Свідоцтво про реєстрацію авторського права на твір № 130631. Комп'ютерна програма «Інформаційна технологія управління відносинами з клієнтами» / Крилик Л. В., Шаргало Д. В., дата реєстрації 14.10.2024 р.
3. Що таке CRM-система та як вона працює. URL: <https://www.creatio.com/ua/crm/what-is-crm> (дата звернення: 27.11.2024).
4. Fatouretchi Max. The Art of CRM: Proven strategies for modern customer relationship management. Packt publishing ltd: BIRMINGHAM-MUMBAI, 2019. 335 р.
5. Salesforce. URL: <https://www.salesforce.com> (дата звернення: 27.11.2024).
6. HubSpot. URL: <https://www.hubspot.com> (дата звернення: 27.11.2024).
7. Zoho CRM. URL: <https://www.zoho.com> (дата звернення: 27.11.2024).
8. Pipedrive. URL: <https://www.pipedrive.com/uk> (дата звернення: 27.11.2024).
9. Що таке JavaScript, та як функціонує JavaScript. URL: <http://ruszura.in.ua/uncategorized/scho-take-javascript-yakfunktsionuyu-javascript.html> (дата звернення: 27.11.2024).
10. JavaScript. URL: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата звернення: 27.11.2024).
11. Development framework. URL: <https://www.3pillarglobal.com/insights/blog/key-decision-criteria-for-selecting-a-development-framework/> (дата звернення: 27.11.2024).
12. NodeJS. URL: <https://nodejs.org/en> (дата звернення: 27.11.2024).
13. Database for your application. URL: <https://wearebrain.com/blog/how-to-choose-the-right-database-for-your-application/> (дата звернення: 27.11.2024).
14. UML diagrams. URL: <https://www.lucidchart.com/blog/types-of-UML-diagrams> (дата звернення: 27.11.2024).

### References

1. Sharhalo D. V Krylyk L. V. Perspektyvy rozrobky informatsiinoi tekhnologii upravlinnia vidnosynamy z kliientamy. Materialy LIH naukovo-tekhnichnoi konferentsii pidrozdiliv Vinnytskoho natsionalnoho tekhnichnoho universytetu, Vinnytsia, 20 – 22 bereznia 2024 r. URL: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2024/paper/view/20386/16891> (дата звернення: 27.11.2024).
2. Svidotstvo pro reiestratsiiu avtorskoho prava na tvir № 130631. Kompiuterna prohrama «Informatsiina tekhnolohiia upravlinnia vidnosynamy z kliientamy» / Krylyk L. V., Sharhalo D. V., data reiestratsii 14.10.2024 r.
3. Shcho take SRM systema ta yak вона пратиує. URL: <https://www.creatio.com/ua/crm/what-is-crm> (дата звернення: 27.11.2024).
4. Fatouretchi Max. The Art of CRM: Proven strategies for modern customer relationship management. Packt publishing ltd: BIRMINGHAM-MUMBAI, 2019. 335 r.
5. Salesforce. URL: <https://www.salesforce.com> (дата звернення: 27.11.2024).
6. HubSpot. URL: <https://www.hubspot.com> (дата звернення: 27.11.2024).
7. Zoho CRM. URL: <https://www.zoho.com> (дата звернення: 27.11.2024).
8. Pipedrive. URL: <https://www.pipedrive.com/uk> (дата звернення: 27.11.2024).
9. Shcho take JavaScript, ta yak funktsionuє JavaScript. URL: <http://ruszura.in.ua/uncategorized/scho-take-javascript-yakfunktsionuyu-javascript.html> (дата звернення: 27.11.2024).
10. JavaScript. URL: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата звернення: 27.11.2024).
11. Development framework. URL: <https://www.3pillarglobal.com/insights/blog/key-decision-criteria-for-selecting-a-development-framework/> (дата звернення: 27.11.2024).
12. NodeJS. URL: <https://nodejs.org/en> (дата звернення: 27.11.2024).
13. Database for your application. URL: <https://wearebrain.com/blog/how-to-choose-the-right-database-for-your-application/> (дата звернення: 27.11.2024).
14. UML diagrams. URL: <https://www.lucidchart.com/blog/types-of-UML-diagrams> (дата звернення: 27.11.2024).