



Приватний заклад вищої освіти

Одеський технологічний університет «ШАГ»

Кафедра інформаційних технологій та фундаментальної
підготовки

випускна кваліфікаційна робота бакалавра

**«ІНФОРМАЦІЙНА СИСТЕМА КЕРУВАННЯ
ЕЛЕКТРОННИМИ ДОКУМЕНТАМИ ТА ПУБЛІКАЦІЯМИ»**

на здобуття бакалаврського рівня вищої освіти
зі спеціальності «122 Комп'ютерні науки»

Виконавці проєкту:

№	П.І.Б.	Група
1	Омельницький М. М.	КН-П-192
2	Желіковський А. В.	КН-П-192
3	Мануйлов М. О.	КН-П-192
4	Горіщак К.С.	КН-Д-192
5	Амікішиєв Е. Н.	КН-А-191

Автор звіту:

_____ Омельницький Микита Миколайович

Одеса – 2023

АНОТАЦІЯ

УДК 004.9

О-57

Омельницький М. М. Інформаційна система керування електронними документами та публікаціями: Автореферат випускної кваліфікаційної роботи на здобуття бакалаврського рівня вищої освіти зі спеціальності «122 Комп'ютерні науки». - Одеса: ОТУШ, 2023 - 17 с.

Ця кваліфікаційна робота присвячена розробці та реалізації інформаційної системи керування електронними документами та публікаціями. В даний час, з поширенням електронних засобів комунікації та збільшенням обсягу електронних документів і публікацій, ефективне управління цими ресурсами стає важливим завданням.

У рамках цього дослідження було проведено аналіз поточного стану керування електронними документами та публікаціями, виявлені основні проблеми і недоліки існуючих систем. На основі цього аналізу були сформульовані вимоги до інформаційної системи, яка повинна бути розроблена.

Розроблено прототип інформаційної системи, що базується на сучасних технологіях керування електронними документами та публікаціями. Ця система надає можливості зберігання, організації та керування електронними документами та публікаціями в уніфікованому середовищі.

Застосування розробленої інформаційної системи може сприяти поліпшенню ефективності роботи з електронними документами та публікаціями, зменшенню часу на пошук та обробку інформації, а також забезпечити більшу надійність та безпеку даних.

У результаті експериментального використання прототипу системи було підтверджено його функціональну придатність та ефективність. Розроблена система може бути застосована в організаціях, які працюють з великим обсягом електронних документів та публікацій, забезпечуючи зручний і безпечний доступ до цих ресурсів.

ЗМІСТ

АНОТАЦІЯ.....	1
ЗМІСТ.....	2
ВСТУП.....	3
ТЕХНІЧНЕ ЗАВДАННЯ ДО ПРОЄКТУ.....	4
РОЗДІЛ 1. Загальна характеристика системи.....	7
РОЗДІЛ 2. Реалізація серверної частини.....	9
РОЗДІЛ 3. Результати випробувань.....	13
ВИСНОВКИ.....	15
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	16

ВСТУП

Сучасний світ знаходиться в епосі цифрової трансформації, де інформаційні технології відіграють вирішальну роль у розвитку суспільства, бізнесу та науки. За останні десятиліття значний прогрес досягнуто в галузі електронного документообігу та публікацій, що призвело до зростання обсягу електронних документів та публікацій, а також до необхідності ефективного керування цими ресурсами.

Інформаційні системи керування електронними документами та публікаціями стають невід'ємною складовою сучасного управління інформацією. Ці системи дозволяють збирати, зберігати, обробляти та розповсюджувати електронні документи та публікації в ефективний і безпечний спосіб. Вони забезпечують централізований доступ до інформації, сприяють її швидкому пошуку та обміну, а також забезпечують збереження та контроль доступу до цінних даних.

Метою даної кваліфікаційної роботи є розробка та аналіз інформаційної системи керування електронними документами та публікаціями, зокрема в контексті використання в наукових організаціях та освітніх установах. Робота спрямована на вивчення сучасних підходів та технологій управління електронними ресурсами, а також розробку пропозицій щодо вдосконалення існуючих систем та процесів.

У процесі дослідження будуть розглянуті основні принципи побудови інформаційної системи керування електронними документами та публікаціями, включаючи збір, зберігання, обробку, пошук та розповсюдження інформації. Будуть досліджені поточні тенденції у цій галузі, а також проблеми, які виникають при впровадженні та використанні інформаційних систем у різних сферах діяльності.

Очікується, що результати цієї роботи допоможуть покращити управління електронними документами та публікаціями, сприятимуть підвищенню ефективності роботи організацій та підприємств, а також сприятимуть подальшому розвитку сфери електронного документообігу та публікацій.

В даному звіті будуть представлені результати дослідження, аналіз отриманих даних, а також рекомендації щодо покращення існуючих інформаційних систем керування електронними документами та публікаціями. Також будуть висунуті пропозиції щодо подальших напрямів досліджень та розвитку даної галузі.

ТЕХНІЧНЕ ЗАВДАННЯ ДО ПРОЄКТУ

Загальний опис

Система управління комерційним контентом призначена для ефективного управління, організації та публікації комерційного контенту в інтернет-проектах, таких як веб-сайти, електронні магазини, блоги і т.д.

Основне призначення системи управління комерційним контентом полягає в тому, щоб забезпечити власникам проєктів і веб-адміністраторам зручні інструменти для створення, редагування, організації, публікації та керування комерційним контентом. Вона допомагає забезпечити актуальність і якість контенту, а також знизити витрати на розробку та обслуговування веб-проєктів.

Призначення

Інформаційна система керування електронними документами та публікаціями призначена для ефективного зберігання, організації, керування та поширення електронних документів та публікацій. Основними цілями такої системи є:

- Збереження електронних документів та публікацій
- Організація та категоризація документів та публікацій
- Керування життєвим циклом документів та публікацій
- Забезпечення безпеки та контролю доступу
- Публікація та поширення інформації

Вимоги до функціоналу

1. Реєстрація та авторизація користувачів:

- Можливість реєстрації нових користувачів з обов'язковим заповненням основних персональних даних.
- Можливість входу в систему для зареєстрованих користувачів за допомогою логіна та пароля або іншого методу авторизації (наприклад, через соціальні мережі).

2. Пошук та навігація:

- Розташування книг за категоріями та жанрами для полегшення пошуку.
- Можливість використовувати різні фільтри для уточнення результатів пошуку (наприклад, за автором, роком видання, видавництвом тощо).

- Результати пошуку повинні бути відображені швидко та ефективно.
3. Відображення публікації:
- Інформація про книгу, включаючи назву, автора, видавництво, опис, обкладинку, рейтинг, відгуки покупців, ціну тощо.
 - Наявність функції "Додати до кошика" або "Придбати" на сторінці товару.
4. Кошик та оформлення замовлення:
- Можливість додавання товарів до кошика, видалення товарів або зміна їх кількості.
 - Відображення загальної суми замовлення та вартості доставки.
 - Обробка замовлення, включаючи введення адреси доставки, способу оплати та перевірку замовлення перед підтвердженням.
5. Оплата та доставка:
- Різні способи оплати, такі як кредитні картки, електронні гроші, оплата при отриманні тощо.
 - Різні варіанти доставки, включаючи доставку поштою, кур'єром, самовивіз з магазину тощо.
 - Відстеження стану доставки та повідомлення про її успішне завершення.
6. Огляди та рейтинги:
- Можливість користувачам залишати відгуки та оцінювати книги.
 - Відображення середньої оцінки книги на її сторінці.
7. Адміністративний функціонал:
- Можливість додавання та видалення книг, оновлення інформації про них адміністратором або продавцем.
 - Керування замовленнями та користувачами.
8. Мобільна сумісність:
- Адаптивний дизайн, що дозволяє зручно переглядати та здійснювати покупки на сайті з мобільних пристроїв.

Архітектура

ПЗ має бути спроектоване за клієнт-серверною архітектурою з розподіленими вузлами:

- Backend - застосунок, що дозволяє керувати даними, зберігати їх у базі даних та віддавати дані за допомогою програмних інтерфейсів (API), якими користуються інші вузли.
- Web Application - застосунок, з яким буде працювати користувач через зрозумілий інтерфейс. Використовує Backend частину для передачі та отримання даних.

Усі модулі ПЗ повинні мати засоби тестування функціональності та розгортання на новому пристрої (інсталяції).

Технології

Зберігання вихідних кодів ПЗ повинно здійснюватись за допомогою системи контролю версій (вибір конкретної системи узгоджується групою проекту). Для кожного з модулів рекомендується (з можливістю часткової зміни) наступні технології створення:

- Backend - JavaScript (Node.js), MongoDB
- Web Application - HTML, CSS, Javascript у складі одного з фреймворків (React.js, Angular.js, Vue.js)

Вихідний код повинен оформлятися у відповідності до загальноприйнятих стандартів, на зразок стандартів GNU <http://www.gnu.org/prep/standards/>, у тому числі з дотриманням вимог до коментування та документування коду.

РОЗДІЛ 1. Загальна характеристика системи

Проект "Книжковий магазин Historium" включає комплексну розробку та реалізацію інформаційної системи для продажу книг. У рамках цього проекту використовуються передові технології та методології для забезпечення ефективного та зручного функціонування системи.

Початково, були створені макети та інтерфейс користувача, використовуючи Figma. При розробці інтерфейсу, було звернуто увагу на розумну навігацію та ергономіку, забезпечуючи комфортне користування системою для різних категорій користувачів.

Клієнтська частина системи була реалізована з використанням фреймворку Vue.js. Це дозволило створити модульну та масштабовану архітектуру, забезпечуючи швидкодію та зручність розробки. Компоненти для головної сторінки, каталогу книг, сторінок окремих книг та кошика покупок забезпечують зручну навігацію та взаємодію з системою.

Серверна частина системи була реалізована з використанням Node.js та Express. Цей підхід дозволив створити потужний та масштабований сервер, який забезпечує обробку запитів та взаємодію з базою даних. API було розроблено для отримання списку книг, деталей книг, додавання товарів до кошика та оформлення замовлення.

Для забезпечення надійності та якості системи, була розроблена тестова частина з використанням Mocha та Supertest. Це дозволило автоматизувати процес тестування та перевірки функціональності системи, забезпечуючи високу стабільність та надійність.

Крім того, для зберігання та керування даними системи була використана база даних MongoDB та фреймворк Mongoose. Це забезпечило зручну роботу з даними, створення схеми та виконання запитів до бази даних.

Одним з важливих аспектів розробки проєкту є використання практик DevOps, що дозволяють забезпечити ефективне співробітництво між розробниками та операторами системи.

Для контролю версій використовується система Git, що дозволяє зберігати, відстежувати та керувати кодом проєкту. Завдяки цьому, команда розробників може працювати одночасно над різними функціональностями та вносити зміни безпеки.

Інфраструктура проєкту розгорнута у хмарних сервісах, що надають гнучкість та масштабованість. Використання хмарних рішень дозволяє ефективно керувати ресурсами та забезпечує високу доступність системи.

Для контейнеризації використовується технологія Docker, що дозволяє пакувати додаток та його залежності у віртуальний

контейнер. Це спрощує розгортання та масштабування системи, забезпечуючи консистентність середовищ та підвищуючи безпеку та надійність.

Автоматизація процесів розробки та розгортання забезпечується за допомогою інструментів Continuous Integration/Continuous Deployment (CI/CD). Це дозволяє автоматично збирати, тестувати та розгортати нові версії додатка, забезпечуючи швидку зміну та впровадження нового функціоналу.

Усі частини системи були інтегровані в єдину цілісну платформу, що дозволяє зручно та ефективно продавати книжки онлайн. Технологічні рішення та використання передових інструментів забезпечують високу продуктивність та надійність системи проєкту "Книжковий магазин Historium".

РОЗДІЛ 2. Реалізація серверної частини

Серверна частина інформаційної системи являє собою Application Programming Interface (далі - API), побудований на архітектурному стилі REST. В якості мови програмування було використано JavaScript в середовищі виконання Node.js з використанням пакетного менеджера NPM. Для створення серверу використано фреймворк Express.js. Додатково було використано:

- **Visual Studio Code** - застосунок для написання коду та управління файловою інфраструктурою проекту
- **Insomnia** - застосунок для написання запитів до API та перевірки їх працездатності
- **Swagger Editor** - веб-застосунок для створення документації до API

По результатам розробки було створено 93 кінцевих точки API, розподілені між 22ма компонентами системи. Кожен компонент описує набір функціональностей (в середньому 4-5), якими користується клієнтська частина за допомогою HTTP запитів. Одними з основних функціональностей є:

- створення користувача
- пошук товару
- додавання позиції до кошику
- оформлення замовлення

Створення користувача здійснюється за допомогою запиту POST /signup. Для створення користувача необхідно передати в тілі запиту такі обов'язкові дані як: ім'я, прізвище, номер телефону, електронна пошта та пароль. Оскільки API побудовано на архітектурному стилі REST, який приймає та віддає дані у форматі JSON, саме в такому форматі клієнтська частина повинна їх передавати:

```
{
  "firstName": "Ім'ян",
  "lastName": "Прізвиськов",
  "phoneNumber": "0672231158",
  "email": "imyan.prizviskov@ukr.net",
  "password": "2yQ5sVJzhM"
}
```

На стороні серверу ці дані проходять валідацію, номер телефону та електронна пошта форматуються за потреби, після чого виконується перевірка існування користувача із вказаним номером телефона або поштою. Далі пароль користувача гешується за допомогою алгоритма SHA256 з використанням солі і дані передаються запиту до бази даних на створення користувача. На останньому кроці клієнтській частині надсилаються дані користувача в тому вигляді, в якому вони зберігаються в базі даних. Пароль та сіль при цьому не передаються:

```
{
  "firstName": "Ім'ян",
  "lastName": "Прізвиськов",
  "phoneNumber": "+380672231158",
  "email": "imyan.prizviskov@ukr.net",
  "role": "user",
  "reviews": [],
  "wishlist": [],
  "history": [],
  "products": [],
  "_id": "64a280c5619d984bcb2b6074",
  "createdAt": 1688371397713,
  "updatedAt": 1688371397713
}
```

Пошук товару здійснюється за допомогою запиту GET /search з параметром запиту q, по значенню якого буде вестися пошук. Приклади такого запиту:

- GET /search?q=Людина - знайде всі товари в назві яких є слово “Людина” (не чутливо до регістру)
- GET /search?q=115977 - знайде товар з кодом 115977

Додавання позиції до кошику здійснюється за допомогою запиту POST /cart-item. Для того щоб запит спрацював необхідно передати заголовок Authorization з авторизаційним токеном користувача та тіло запиту з інформацією про позицію:

```
{
  "product": "6473b317569debe2438c7856",
  "quantity": 3
}
```

Після потрапляння запиту до обробника перевіряється існування вказаного товару та корзини користувача, яка вилучається з отриманого авторизаційного токена і, якщо позиція з таким товаром вже існувала в корзині оновлюється її кількість, інакше - створюється нова.

Оформлення замовлення є відносно великим функціоналом і здійснюється за допомогою запиту POST /order. Обов'язковим є тіло запиту, в якому передається вся інформація про замовлення:

```
{
  "contactInfo": {
    "firstName": "Ім'ян",
    "lastName": "Прізвиськов",
    "phoneNumber": "+380442138972",
    "email": "imyan.prizviskov@ukr.net"
  },
  "receiverInfo": {
    "firstName": "Прізва",
    "lastName": "Ім'янова",
    "phoneNumber": "+380445139822"
  },
  "companyInfo": {
    "name": "Компакт",
    "identificationNumber": "18452271",
    "address": "вул. Поточна, 23/1"
  },
  "callback": true,
  "deliveryInfo": {
    "country": "Україна",
    "city": "Полтава",
    "type": "Відділення Нова Пошта",
    "address": "просп. Довідкова, 12"
  },
  "paymentType": "Готівкою або картою: При отриманні",
  "items": [
    {
      "product": "6473c4ef569debe2438c794f"
    },
    {
      "product": "6474dd020472d62ed62f4513",
      "quantity": 3
    }
  ]
}
```

Опціонально запит може містити заголовок Authorization із авторизаційним токеном користувача. Залежно від вхідних даних результат такого запиту можна поділити на 3 різновиди:

- Передано контактну інформацію неіснуючого користувача, без токена. В результаті такого запиту буде створено нового користувача на основі контактних даних і надіслано 2 листи: в одному тимчасовий пароль для доступу в акаунт, в іншому номер замовлення
- Передано контактну інформацію існуючого користувача, без токена. В результаті такого запиту клієнтська частина отримає повідомлення, що для оформлення такого замовлення користувач повинен бути авторизованим
- Передано контактну інформацію існуючого користувача, з токеном. В результаті такого запиту корзину користувача буде очищено та надіслано листа з номером замовлення

Фрагмент коду, відповідний за надсилання листа у разі створення нового користувача:

```
const password = randomBytes(8).toString('hex');

const mailData = {
  from: '"Historium" noreply@historium.store',
  to: email,
  subject: 'Реєстрація',
  html: `Тимчасовий пароль: <b>${password}</b>`
};

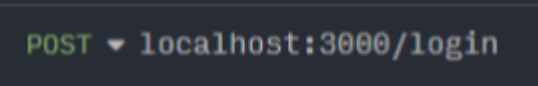
await transporter.sendMail(mailData);
```

РОЗДІЛ 3. Результати випробувань

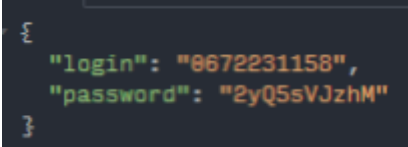
Для того щоб сервер запрацював необхідно вказати змінні оточення, які зберігають інформацію про порт на якому сервер буде працювати, строку підключення до бази даних та інші важливі змінні. Вказуються ці змінні по-різному в залежності від середовища на якому запускається сервер: або в dockerfile (для виробничого середовища), або через NPM пакет dotenv (для середовища розробки).

Після того як сервер запустився, він готовий приймати запити. Для того щоб до API могла звертатися тільки клієнтська частина в виробничому середовищі вона повинна в кожному запиті надсилати заголовок API-Key, значення якого буде перевірятися першим ділом при потраплянні запиту на сервер. Нижче наведено приклади деяких запитів та результати їх виконання:

- Авторизація користувача:



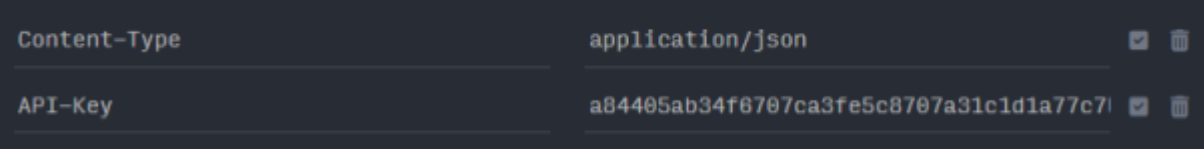
POST localhost:3000/login



```
{
  "login": "8672231158",
  "password": "2yQ5sVJzhM"
}
```

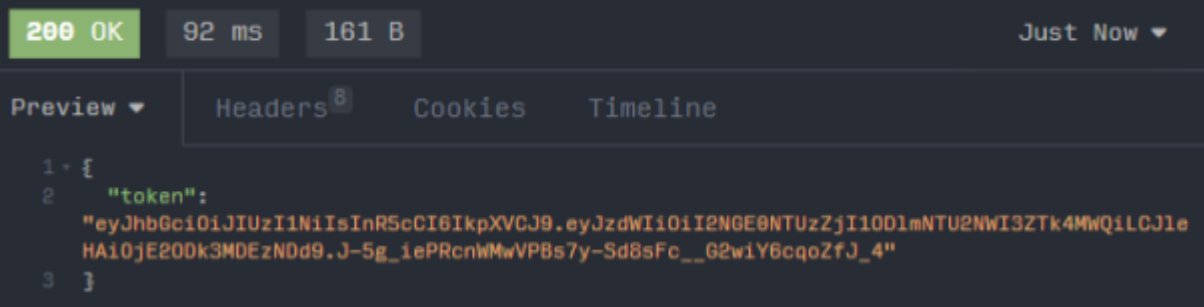
Строка запиту

Тіло запиту



Content-Type	application/json	✕	🗑
API-Key	a84405ab34f6707ca3fe5c8707a31c1d1a77c7f	✕	🗑

Заголовки запиту



200 OK 92 ms 161 B Just Now

Preview Headers Cookies Timeline

```
1 {
2   "token":
3     "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI2NGE0NTUzZjI1OD1mNTU2NWl3ZTk4MWwQlCJleHA10jE2ODk3MDEzNDd9.J-5g_iePRcnWmVP8s7y-Sd8sFc__G2wiY6cqoZfJ_4"
```

Результат запиту

- Оформлення замовлення з існуючим користувачем:

POST ▾ localhost:3000/order

Строка запиту

Content-Type	application/json	🗕 🗑
API-Key	a84405ab34f6707ca3fe5c8707a31c1d1a77c7b6976fa	🗕 🗑
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e	🗕 🗑

Заголовки запиту

```
{
  "contactInfo": {
    "firstName": "Ім'я",
    "lastName": "Прізвиський",
    "phoneNumber": "0672231158",
    "email": "imyan.prizviskov@ukr.net"
  },
  "receiverInfo": {
    "firstName": "Прізвиський",
    "lastName": "Ім'я",
    "phoneNumber": "380445139822"
  },
  "companyInfo": {
    "name": "Компакт",
    "identificationNumber": "18452271",
    "address": "вул. Поточна, 23/1"
  },
  "callback": true,
  "deliveryInfo": {
    "country": "Україна",
    "city": "Полтава",
    "type": "Відділення Нова Пошта",
    "address": "просп. Довідкова, 12"
  },
  "paymentType": "Готівкою або карткою: При отриманні"
}
```

Тіло запиту

```
{
  "contactInfo": {
    "firstName": "Ім'я",
    "lastName": "Прізвиський",
    "phoneNumber": "+380672231158",
    "email": "imyan.prizviskov@ukr.net"
  },
  "receiverInfo": {
    "firstName": "Прізвиський",
    "lastName": "Ім'я",
    "phoneNumber": "+380445139822"
  },
  "gift": false,
  "companyInfo": {
    "name": "Компакт",
    "identificationNumber": "18452271",
    "address": "вул. Поточна, 23/1"
  },
  "callback": true,
  "deliveryInfo": {
    "country": "Україна",
    "city": "Полтава",
    "type": "Відділення Нова Пошта",
    "address": "просп. Довідкова, 12"
  },
  "paymentType": "Готівкою або карткою: При отриманні",
  "status": {
    "name": "Поточний",
    "key": "active"
  },
  "user": { ↩ 15 ↩ },
  "items": [ ↩ 2 ↩ ],
  "totalPrice": 1320,
  "totalQuantity": 4,
  "_id": "64a45ab32589f5565b7e9846",
  "createdAt": 1688492723883,
  "updatedAt": 1688492723883,
  "number": "2000134353"
}
```

Результат запиту

Таким чином результати запитів повертають очікувану інформацію та відповідні статус-коди, які використовуються клієнтською частиною для виконання певних дій в залежності від статус-кода.

ВИСНОВКИ

Проведено огляд інформаційних систем керування електронними документами та публікаціями. Виявлено, що суттєве поширення мають системи, пов'язані з комерційною діяльністю. В якості такої системи було обрано та реалізовано інтернет-магазин по продажу книг.

На етапі проектування було досліджено декілька існуючих інтернет-магазинів на предмет функціоналів, їх актуальність, можливі альтернативи та покращення. Основою дослідження став інтернет-магазин Yakaboo (<https://www.yakaboo.ua>). В результаті проектування було описано 22 сутності для програмної реалізації у вигляді компонентів.

Серверну частину проекту було вирішено будувати у вигляді API на архітектурному стилі REST. Такий стиль було обрано через гарну сумісність та можливість виконання поставлених задач, його розповсюдженість, а також актуальність і простоту.

В результаті розробки було створено та задокументовано 94 кінцевих точки API, кожна з яких дозволяє проводити операції над певним компонентом. Таким чином 15 з 22 компонентів мають повний набір Create, Read, Update, Delete (CRUD) операцій, інші 7 мають частковий набір через відсутність необхідності в повному.

Для забезпечення безпеки персональних даних користувачів було впроваджено хешування паролів за допомогою алгоритма SHA256 з додаванням солі. Також створено можливість відновлення паролю шляхом відправки тимчасового на електронну пошту або телефон користувача. Тимчасовий пароль дозволяє користувачу авторизуватися до свого акаунту та зберігається доки користувач не оновить старий.

Систему було випробувано шляхом написання запитів у застосунку Insomnia, а також шляхом її використання клієнтською частиною.

Підсумовуючи можна сказати, що впроваджені засоби відповідають технічному завданню і можуть бути доповнені та покращенні у майбутньому. Система створювалась маючи на увазі певний рівень абстракції, щоб залишити можливість її подальшого розширення.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Стратегія розвитку інформаційного суспільства в Україні. Розпорядження Кабінету Міністрів України від 15 травня 2013 р. № 386-р. // Урядовий портал. URL: <https://www.kmu.gov.ua/npas/246420577>
2. Стратегія кібербезпеки України. Указ Президента України від 14 травня 2021 року // Урядовий кур'єр від 28.08.2021 — № 165. URL: <https://zakon.rada.gov.ua/laws/show/447/2021/print>
3. Решетило В.П., Федотова Ю.В. Невизначеність та ризик: співвідношення понять та специфіка прийняття рішень // Проблеми системного підходу в економіці, випуск № 3(77)-2, 2020. с. 149. URL: http://psae-jrnl.nau.in.ua/journal/3_77_2_2020_ukr/23.pdf
4. Олександр Смичніков. Параноя або сучасні реалії: як зберегти конфіденційність у мережі. - ФЬЮЧЕР МЕДІА. URL: <https://mind.ua/openmind/20183329-paranoya-abo-suchasni-realiyi-yak-zberegiti-konfidencijnist-u-merezhi>
5. Анісімов А.В. Інформаційні системи та бази даних: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. / Анісімов А.В., Кулябко П.П. - Київ. - 2017. - 110 с.
6. Величко О.М. Інтелектуальні інформаційні системи: структура і застосування. Підручник / Величко О.М., Гордієнко Т.Б. - Херсон. - 2022. - 728 с. ISBN: 978-966-289-552-0
7. Булгакова О.С. Інформатика: візуальне програмування. Навчальний посібник (стереотипне видання) / Булгакова О.С., Зосімов В.В., Броницька Н.А., Танкова Н.В. - Херсон. - 2020. - 312 с. ISBN: 978-966-289-039-6
8. Покотилова В.І. Використання інформаційних технологій в теорії прийняття рішень. Навчальний посібник / Покотилова В.І., Фомішина В.М., Лугінін О.Є. - Херсон. - 2019. - 240 с. ISBN: 978-966-289-277-2
9. Величко О.М. Основи системного аналізу і прийняття оптимальних рішень. Підручник / Величко О.М., Гордієнко Т.Б. - Херсон. - 2021. - 672 с. ISBN: 978-966-289-553-7
10. Чайковська М.П. Концептуально-методологічні засади управління маркетинговими ІТ-проектами в умовах цифрових трансформацій. Монографія / Чайковська М.П. - Херсон. - 2021. - 370 с. ISBN: 978-966-289-537-7