



Приватний заклад вищої освіти
Одеський технологічний університет «ШАГ»
Кафедра інформаційних технологій та фундаментальної підготовки

АВТОРЕФЕРАТ
випускної кваліфікаційної роботи бакалавра
**«СИСТЕМА УЗГОДЖЕННЯ ВЗАЄМОДІЇ КОРИСТУВАЧІВ
З УРАХУВАННЯМ ГЕОПОЗИЦІОНУВАННЯ»**

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

Виконавці проєкту

№	П.І.Б.	Група
1	Богдан Д. М.	КН-П-181
2	Боженівич Д. С.	КН-Д-182
3	Демиров Д. А.	КН-П-181
4	Казанцев А. В.	КН-Д-182

Автор звіту

_____ Богдан Денис Максимович

РЕЗУЛЬТАТИ ЕКСПЕРТИЗИ

Експерт	П.І.Б.	Оцінка	Підпис
Науковий керівник	доц. Самойленко Д. М.	В (86)	
Рецензент	доц. Суліма М. М.	В (82)	
ДЕК (захист проєкту)	доц. Голова Е. К.	В (84)	

АНОТАЦІЯ

УДК 004.9

Б-73

Богдан Д. М. Система узгодження взаємодії користувачів з урахуванням геопозиціонування: Автореферат випускної кваліфікаційної роботи на здобуття бакалаврського рівня вищої освіти зі спеціальності «122 Комп'ютерні науки». - Одеса: ОТУШ, 2022 - 16 с.

Робота присвячена дослідженням, розвиненню та впровадженню інформаційної системи, головним призначенням якої є узгоджена взаємодія між користувачами з урахуванням їх географічної локації. Додаткові функції системи включають реєстрацію, авторизацію та автентифікацію, засоби обмеження доступу до контенту, ведення журналів активності користувачів системи.

Актуальність роботи визначається поширеністю задач, які ґрунтуються на оперативній фільтрації результатів за географічною ознакою, зокрема, за віддаленістю від поточного місцезнаходження об'єкта, а також необхідністю врахування у складі комплексної оцінки волатильності результату місцезнаходження суб'єкта пошуку. Особливу увагу звертає на себе той факт, що засоби встановлення локації можуть мати суттєві похибки або відмови спрацювання. Це дозволяє класифікувати задачі як такі, що формуються в умовах невизначеності або неповної визначеності вихідних даних. Додатково, актуальність диктується потребою впровадження мобільних інформаційних систем та реалізації у них покращених засобів захисту інформації, зокрема, персональних даних

Об'єктом дослідження у рамках даної роботи виступає алгоритм складання комплексної оцінки результатів за багатьма критеріями в умовах неповної визначеності вихідних умов. Предметом дослідження обрано фактор географічної позиції (локації) користувача на момент формування запиту. У якості задач дослідження було встановлене наступне:

- провести огляд інформаційних систем, що враховують відомості про локацію користувачів, встановити характерні риси та напрями удосконалення.
- визначити програмні засоби (інтерфейси API), які дозволяють отримувати дані про локацію користувача, встановити їх особливості та ступінь вживаності для поставлених завдань.
- скласти програмне забезпечення, реалізувавши у ньому результати, досягнуті у попередніх задачах, провести випробування за різних умов та для різних пристроїв.
- впровадити у систему програмні засоби захисту даних для покращення показників інформаційної безпеки створеної системи.

Методи дослідження, що використані для досягнення поставлених задач, утворені суперпозицією методів системного аналізу, моделювання та когнітивної методології.

При складанні звіту використано 42 посилання на друковані та електронні джерела інформації.

ЗМІСТ

ВСТУП _____	3
Технічне завдання до проєкту _____	4
РОЗДІЛ 1. Загальна характеристика системи _____	6
РОЗДІЛ 2. Реалізація серверної частини _____	7
РОЗДІЛ 3. Результати випробувань _____	11
ВИСНОВКИ _____	14
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ _____	15

ВСТУП

Розвиток і поширення інформаційних технологій супроводжується перенесенням великої кількості послуг у цифровий простір. Стратегія розвитку інформаційного суспільства в Україні визначає, що «комунікаційні технології є необхідним інструментом соціально-економічного прогресу, одним з основних чинників інноваційного розвитку економіки» [1].

Сучасні мобільні пристрої, - смартфони, планшети, ноутбуки, - практично всі обладнані пристроями визначення геолокації, що значно покращує ефективність прикладного програмного забезпечення, яке здатне врахувати у своїй роботі дані щодо розміщення користувача. Подібні технології широко проникають у всі галузі діяльності, створюючи окремі напрями наукового дослідження - ГІС (геоінформаційні системи) [2].

У той же час, широкий вжиток інформаційних систем, особливо з реалізованими засобами локації, потребує покращення інформаційної безпеки як щодо захисту персональних даних користувача, так і щодо відомостей про його місцезнаходження [3]. Це актуалізує додаткові дослідження і розробки в зазначеній області.

В якості предмету даної роботи було обрано досить поширену систему узгодження взаємодії користувачів до функцій якої належать розміщення пропозицій, що надходять з боку одних користувачів, та пошук відповідних пропозицій з боку інших. При пошуку має враховуватись обмеження, що їх накладають географічні локації обох користувачів.

Також функції переважної більшості подібних системи мають передбачати можливість збереження історії розміщення, пошуку пропозицій, а також фактів їх прийняття чи відхилення. У перспективі - засобів оплати. Відповідно, це передбачає засоби авторизації та автентифікації, а також убезпечення персональних даних від несанкціонованого доступу інших користувачів.

Детальне технічне завдання до проєкту наводиться далі. У рамках даного звіту розглядається серверна частина, що являє собою ПЗ, створене на платформі NodeJS за допомогою мови JavaScript. Звіт складається з трьох розділів у яких описується загальна характеристика системи, деталі створення серверної частини та результати її розгортання та тестування. Висновки відбивають результати, що досягнуті у рамках даної частини проєкту, загальний висновок є сукупністю звітів усіх учасників проєкту.

ТЕХНІЧНЕ ЗАВДАННЯ ДО ПРОЄКТУ

Загальний опис.

Програмне забезпечення (ПЗ) призначене для створення розподіленої системи узгодження взаємодії користувачів, що поділяються за різними ролями. Головна функція ПЗ полягає в узгодженні інтересів користувача, який надає послуги, та користувача, що споживає послугу. Додаткова функція ПЗ має дозволити залишати відгуки та оцінки щодо усіх видів користувачів

Призначення:

ПЗ орієнтується на різні види узгодження користувачів: за географічним розташуванням, за видом послуги, за вартістю, за рейтингом тощо. ПЗ створюється за прикладом замовлення транспортних послуг, у той же час має складатись за принципом універсальності, мати змогу змінити цільове призначення.

Вимоги до функціоналу:

Реєстрація користувачів:

Користувач ПЗ має змогу зареєструватись як за допомогою авторизаційних даних (логіну/паролю), так і за допомогою облікових записів популярних соціальних мереж (Google, Facebook, тощо).

При реєстрації обов'язково зазначається засіб оперативного контактування (ЗОК) - телефон. На тестовому етапі, за умови ускладнення замовлення послуг телефонії (та/або СМС) можливе використання електронної пошти. Засіб оперативного контактування при реєстрації підлягає верифікації через відповідь на тестовий виклик (лист) або введення коду, пересланого через повідомлення. Рекомендується використовувати ЗОК у якості логіну користувача.

Також при реєстрації зазначається роль користувача: споживач або надавач послуги. Одна особа, використовуючи одні авторизаційні дані, може зареєструватись і як споживач, і як надавач послуги.

Для можливості ввічливого спілкування між користувачами при реєстрації має зазначатись ім'я, за яким до користувача будуть звертатись. Відповідне повідомлення (про те, що зазначене ім'я буде у листуванні) має бути розміщене на сторінці реєстрації.

Особистий кабінет надавача послуги:

Основне робоче місце користувача ПЗ. Повинен забезпечити усі необхідні засоби для

- Розміщення пропозицій
- Встановлення обмежень щодо рейтингу споживача
- Перегляд історії

Особистий кабінет споживача послуги

Так само, являє собою основний робочий засіб. Має реалізовувати можливості

- Розміщення замовлень
- Пошук пропозицій (оперативні замовлення у реальному часі)
- Перегляд відгуків та рейтингу

Картографія

ПЗ має підтримувати засоби ГІС (геоінформаційних систем) для спрощення локалізації надавача і замовника послуги. ПЗ має запитати дозвіл на визначення поточного місця знаходження пристрою, з якого користувач застосовує ПЗ. При розміщенні пропозиції (замовленні) користувач повинен мати можливість зазначити власне місце на інтерактивній карті автоматично або у ручному режимі. Аналогічну можливість повинен мати користувач - надавач послуги.

У режимі пошуку пропозицій або замовлень користувачі ПЗ повинні мати режим перегляду результатів пошуку на карті з можливістю одержати деталі кожного зі знайдених об'єктів через карту (вибір вказівником миші або сенсором)

Платіжний засіб (гаманець)

Задля спрощення взаємних розрахунків ПЗ повинне реалізовувати функції гаманця. Дана функція забезпечує накопичення "бонусів" (віртуальних грошей), за допомогою яких можна здійснювати замовлення послуг. У перспективі передбачається можливість придбання бонусів через стандартні платіжні системи. ПЗ має забезпечувати можливість обмеження доступу користувачів до послуг за умови недостатньої кількості бонусів, а також встановлення обмежень при розміщенні пропозицій щодо стану гаманця замовника.

Зворотній зв'язок

За фактом завершення взаємодії надавача та споживача послуги ПЗ пропонує обом сторонам надати відгук та скласти оцінку якості іншої сторони. Оцінки та відгуки мають відображатись через особистий кабінет, а також бути доступними іншим користувачам ПЗ для ознайомлення.

Архітектура

ПЗ має бути спроектоване за архітектурою PWA (Progressive Web Application) з розподіленими вузлами:

Backend - серверне програмне забезпечення, що забезпечує збереження та оброблення поточних даних, взаємодію з базою даних. Надає програмний інтерфейс взаємодії (API) з іншими вузлами

WebApp - додаток, доступний з веб-простору (сайт), що забезпечує візуальний інтерфейс користувачів усіх категорій

MobileApp - додаток, що встановлюється на мобільних пристроях (смартфон, планшет), додатково до WebApp забезпечує моніторинг замовлень у фоновому режимі, дозволяє накопичувати дані при відсутності мережного з'єднання (офлайн) з подальшою синхронізацією з Backend

.

Усі модулі ПЗ повинні мати засоби

- тестування функціональності,
- розгортання на новому пристрої (інсталяції)

Технології

Зберігання вихідних кодів ПЗ повинно здійснюватись за допомогою системи контролю версій (вибір конкретної системи узгоджується групою проекту). Структура файлів та каталогів повинна відбивати архітектуру ПЗ, на зразок:

- root
 - Backend
 - Web
 - Mobile

Для кожного з модулів рекомендується (з можливістю часткової зміни) наступні технології створення:

Backend - PHP, MySQL

Web - JS, HTML, CSS, або JS фреймворки, зокрема ReactJS, AngularJS

Mobile - Java, або Kotlin, або Flutter

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

РОЗДІЛ 1

ЗАГАЛЬНА ХАРАКТЕРИСТИКА СИСТЕМИ

У відповідності до технічного завдання (ТЗ) система була спроектована за архітектурою «Progressive Web Application» (PWA). У складі системи було умовно відокремлено наступні архітектурні частини:

- «Mockup»,
- «Backend»,
- «Frontend».

Частина «Mockup» являє собою набір моделей та описів поведінки користувача. Модель подає інтерфейси взаємодії користувачів різного типу з інформаційною системою, траєкторії рухів та переходів між різними інтерфейсами. При створенні інтерфейсів застосовано принципи розумної навігації, включно із підходами UI/UX, висновки колористики та ергономіки мобільних пристроїв. Частина розміщена за адресою <https://www.figma.com/file/gDr7qqU2ZRd0wt3giXqOnf/Untitled?node-id=0%3A1>.

Частина «Backend» реалізована як централізований вузол збереження та оброблення даних, що створюються та модифікуються протягом життєвого циклу інформаційної системи. Дана частина включає у себе базу даних проєкту та програмний інтерфейс (API) доступу до основних її функцій, зокрема, керування контентом, авторизації, узгодження профілів користувачів різного типу та/або різних пристроїв. Частина «Backend» виконана на базі програмного забезпечення NodeJS та СУБД «MySQL». Додатково використані програмні пакети «Express», «Mysql2». Частина розміщена за адресою https://bitbucket.org/bohdan_denys/taxi-website/src/master/backend/

Клієнтська частина системи «Frontend», призначена для надання користувачеві доступу до функцій системи через засоби візуального інтерфейсу. Також до функцій даної частини належить збереження локальних даних для можливості часткового використання системи без мережі передавання даних (offline). За наявності у пристрої користувача модулів визначення локації, засобами даної частини мають бути передбачені функції одержання та оброблення їх сигналів. Вихідні коди та ресурси даної частини розміщені на репозиторії https://github.com/KristarDen/Taxi_android_app.git. Подальший звіт присвячений детальному опису цієї частини.

З метою покращення безпеки, звертаючи увагу на той факт, що мобільні пристрої досить часто підключаються до незахищених мереж передавання даних, інформаційний обмін між частинами «Backend» та «Frontend» захищений попереднім шифруванням сучасним криптоалгоритмом AES-256. Це забезпечує роботу системи у відкритих мережах, а також спрощує задачі автентифікації та доступу до контенту з обмеженим доступом без наявності ключів розшифрування даних, навіть за умови дискредитації інших модулів системи.

РОЗДІЛ 2

РЕАЛІЗАЦІЯ СЕРВЕРНОЇ ЧАСТИНИ

Серверна частина інформаційної системи реалізована з використанням наступних технологій:

- Засоби розроблення: мова програмування JavaScript та редактор Visual Studio Code
- Додаткове ПЗ та засоби:
 - Express, як базовий каркас для побудови серверної частини.
 - MySQL2 для роботи з базою даних.
 - Модуль NodeMailer для підтримки зв'язку з користувачем через електронну пошту.
 - Модуль Crypto для реалізації шифрування даних.
 - Модуль cryptoRandomString для власного створення ідентифікаторів користувачів та кодів підтвердження.

У складі серверної частини передбачено та реалізовано наступні функції:

- Реєстрація користувача і внесення його даних у БД
- Підтвердження даних шляхом надсилання коду на пошту
- Авторизація користувача
- Додавання фотографії профілю
- Створення замовлення та його публікація
- Отримання історії поїздок

Початок роботи користувача із додатком супроводжується його реєстрацією як учасника інформаційної системи. При реєстрації користувач обов'язково надає відомості про електронну пошту, через яку буде здійснюватись оповіщення користувача щодо актуальних питань, а також про зміну активності у його особистому просторі. Попередньо до реєстрації користувача проводиться пошук за введеними даними у БД для уникнення ситуації повторної реєстрації користувача.

```
let userExists = await dbUser.getUserByPhoneNumber(phone);

if(userExists.length > 0) {
    returnData = crypt.createCipher(JSON.stringify({message: 'The user is
already registered by this phone number'}));
    return res.status(400).json({data: returnData})
}
```

Авторизація користувача здійснюється за допомогою персонального секрету (логіну та паролю), введеного на етапі реєстрації.

При реєстрації чи авторизації здійснюється комунікація з частиною «Frontend», яка надсилає дані на сервер для їх подальшої валідації, у результаті чого взаємодія підтверджується або спростовується. Для прикладу наведено опрацювання даних при авторизації:

```

router.post('/login', async(req, res) => {
  try {
    const { phone, password } =
JSON.parse(crypt.createDecipher(req.body.data))

    let user = await dbUser.getUserByPhoneNumber(phone);
    if(user.length === 0)
    {
      returnData = crypt.createCipher(JSON.stringify({message: 'User not
found'}));
      res.status(400).json({data: returnData});
    }
    if(crypt.createCipher(password) === user[0].password)
    {
      const token = generateAccessToken(user[0].id);
      returnData = crypt.createCipher(JSON.stringify({token: token}));
      return res.json(returnData);
    } else {
      returnData = crypt.createCipher(JSON.stringify({message: 'User
not found'}));
      res.status(400).json({data: returnData});
    }
  } catch(err) {
    returnData = crypt.createCipher(JSON.stringify({message: 'Something
went wrong, try again later'}));
    res.status(500).json({data: returnData});
    console.log('Server error on /auth/login', err.message);
  }
})

```

Опрацювання даних реалізовано за допомогою окремих програмних засобів та модулів. У якості прикладу далі наведено приклад опрацювання даних при реєстрації користувача та підтвердження його особистості за допомогою програмно генерованого коду.

```

function addUser(phone, name, surname, email, password, userCode) {
  const id = cryptoRandString({length: 12, type: 'numeric'});
  const sql = 'INSERT INTO users (id, phone, name, surname, email,
password, confirmed) VALUES (?, ?, ?, ?, ?, ?, ?)';
  const data = [id, phone, name, surname, email, password, userCode];

  return new Promise((resolve, reject) => {
    pool.query(sql, data, function(err, results){

```

```

        if(err) reject(err);
        console.log(`Inserted user with phone: ${phone}`);
        resolve(results);
    })
})
}

function confirmCode(code) {
    const sql = 'UPDATE confirmation_codes SET confirmed = true WHERE code = ?';
    const data = [code];
    return new Promise((resolve, reject) => {
        pool.query(sql, data, function(err, results){
            if(err) reject(err);
            console.log('Verification code confirmed');
            resolve(results);
        })
    })
}

```

Основну частину взаємодії серверної частини та клієнтської складає опрацювання даних, які надсилає чи отримує користувач для їх подальшого відображення чи збереження. Клієнтська частина надсилає запити на отримання, запис чи редагування даних, які потрібні користувачу за допомогою методів описаних у частині Frontend.

Для забезпечення описаної функціональності у проєкті вжито наступні засоби: MySQL2 – підключення до бази даних проєкту, опрацювання даних БД; CryptoRandomString - для створення ідентифікатору записів та унікальних кодів підтвердження особистості; Crypto – шифрування критично важливих даних користувачів за допомогою алгоритму AES-256 (режим CBC), який є стандартом в багатьох країнах та є одним з надійніших алгоритмів шифрування.

Для зручності функціонал цих модулів розподілені у різні файли та використовуються окремими функціями, що зображено нижче. Для прикладу наведено опис функціоналу модуля Crypto, що шифрування та дешифрування усіх даних, які проходять через серверну частину.

```

function createCipher(data) {
    try {
        let cipher = crypto.createCipheriv(algorithm, key, iv);

        let encryptedData = cipher.update(data, 'utf-8', 'base64')
        encryptedData += cipher.final('base64');
        console.log("Encrypted data: ", encryptedData);
        return encryptedData;
    } catch(err) {
        console.log(`Server error! The encryption method returned an error ${err}`);
    }
}

```

```

    }
}

function createDecipher(data) {
  try {
    const decipher = crypto.createDecipheriv(algorithm, key, iv);

    let decryptedData = decipher.update(data, 'base64', 'utf-8')
    decryptedData += decipher.final('utf-8');

    console.log("Decrypted data: ", decryptedData);
    return decryptedData;
  } catch(err)
  {
    console.log(`Server error! The decryption method returned an error
    ${err}`);
  }
}

module.exports = {createCipher, createDecipher};

```

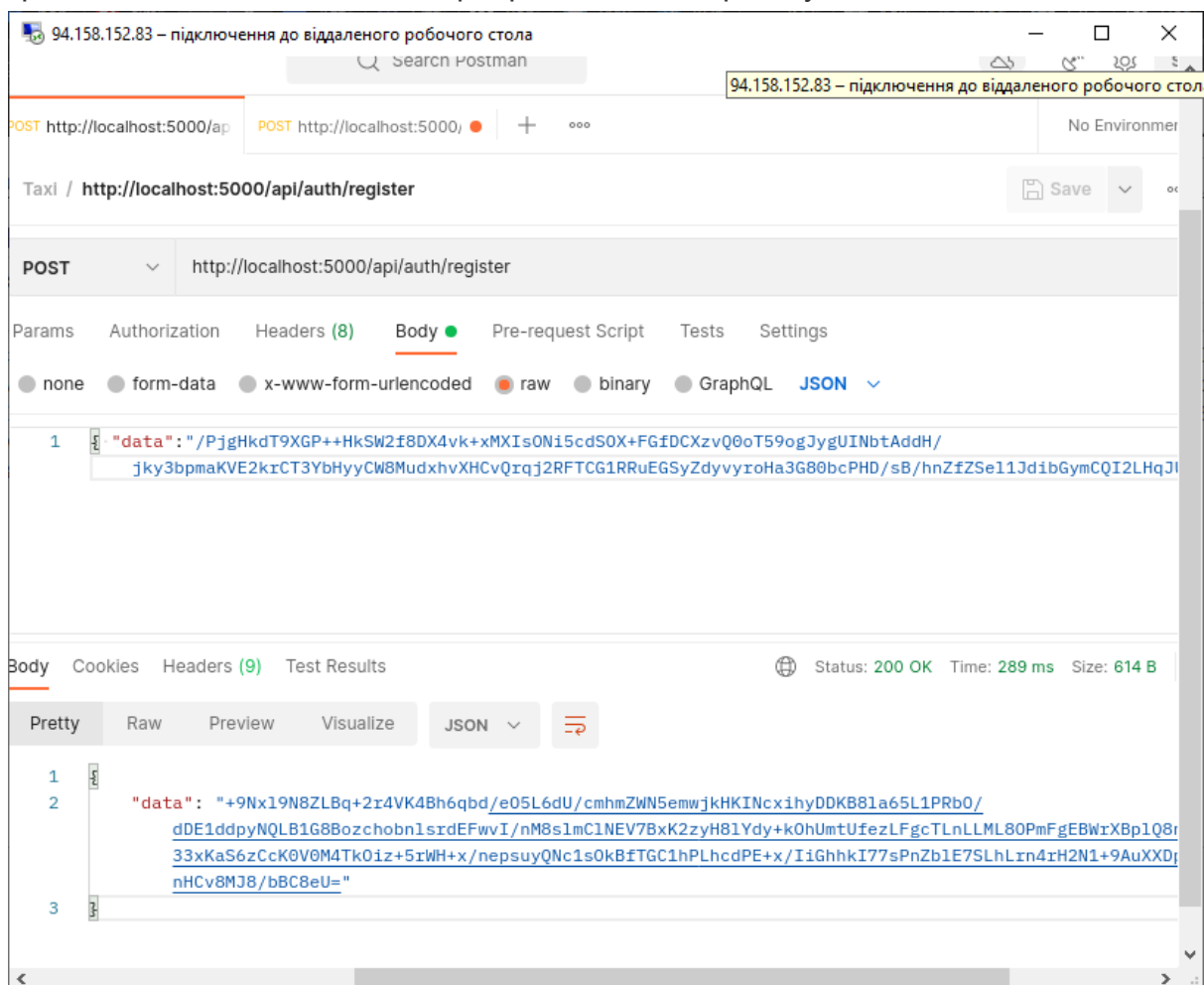
Як вже зазначалось вище, з повним кодом проєкту можна ознайомитись у репозиторії https://bitbucket.org/bohdan_denys/taxi-website/src/master/backend/

РОЗДІЛ 3 РЕЗУЛЬТАТИ ВИПРОБУВАНЬ

Результатом компіляції програмного коду серверної частини являється сервер, який працює на окремій фізичній машині. Головна серія випробувань, звіт з якої наведено далі, проведена за допомогою допоміжного ПЗ «Postman».

Початок роботи нового користувача з додатком починається з реєстрації у системі. Під час реєстрації клієнтська частина надсилає на сервер зашифровані дані, який увів користувач. Після опрацювання цих даних сервер, в залежності від результату перевірки, або реєструє нового користувача, надсилає код на пошту для підтвердження та у відповідь користувачу відправляє токен. Або надсилає у відповідь повідомлення про існуючого користувача.

На рис. 1 зображено вхідні дані та відповідь серверу на них, а також на спробу повторної реєстрації. Також тут видно приклад того, як шифруються дані при взаємодії клієнтської та серверної частини проєкту.



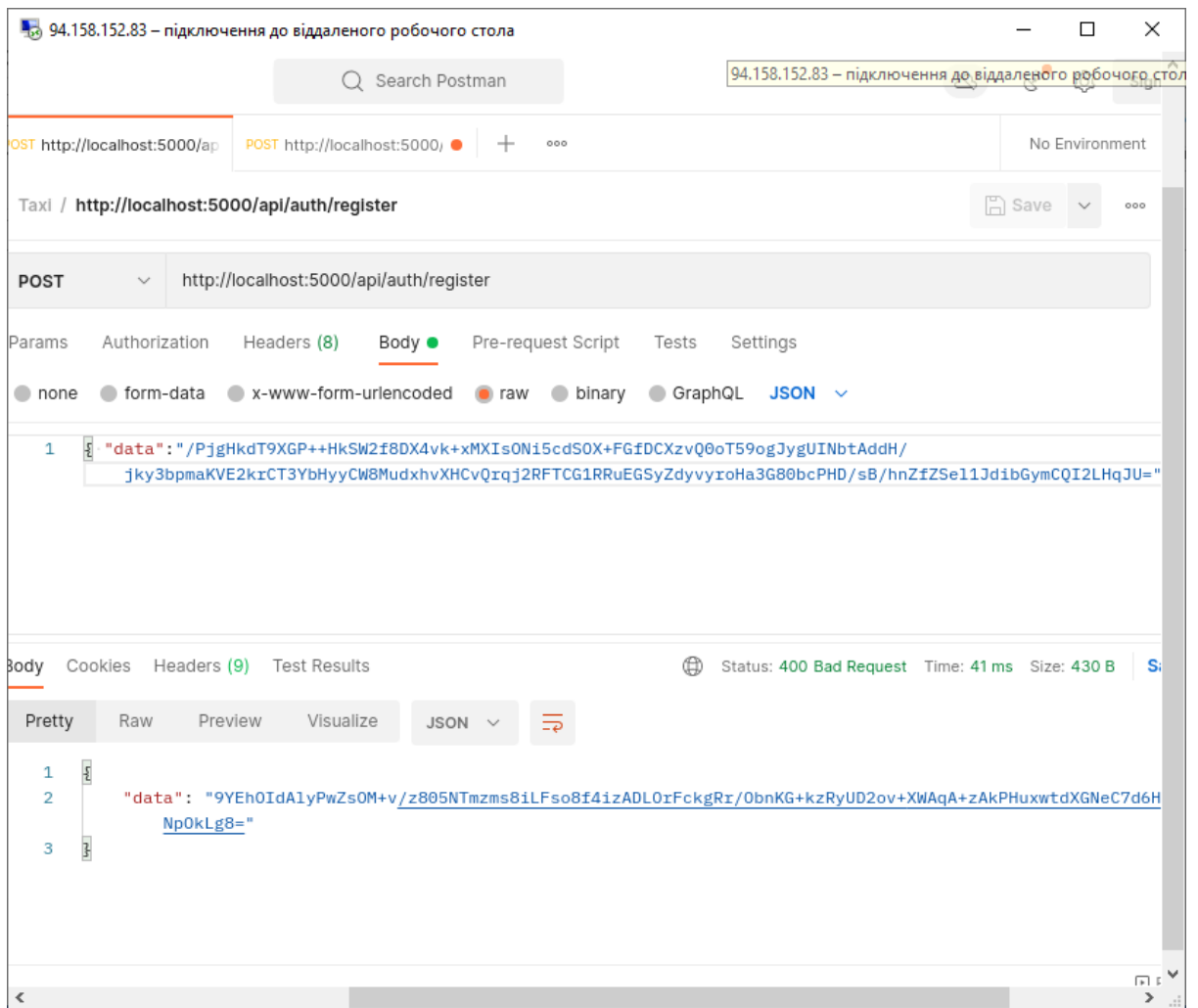


Рисунок 1 – Вхідні та вихідні дані реєстрації, спроба повторної реєстрації

На рис. 2 показано надісланий код на пошті користувача та відповідь сервера при його надсиланні на перевірку.

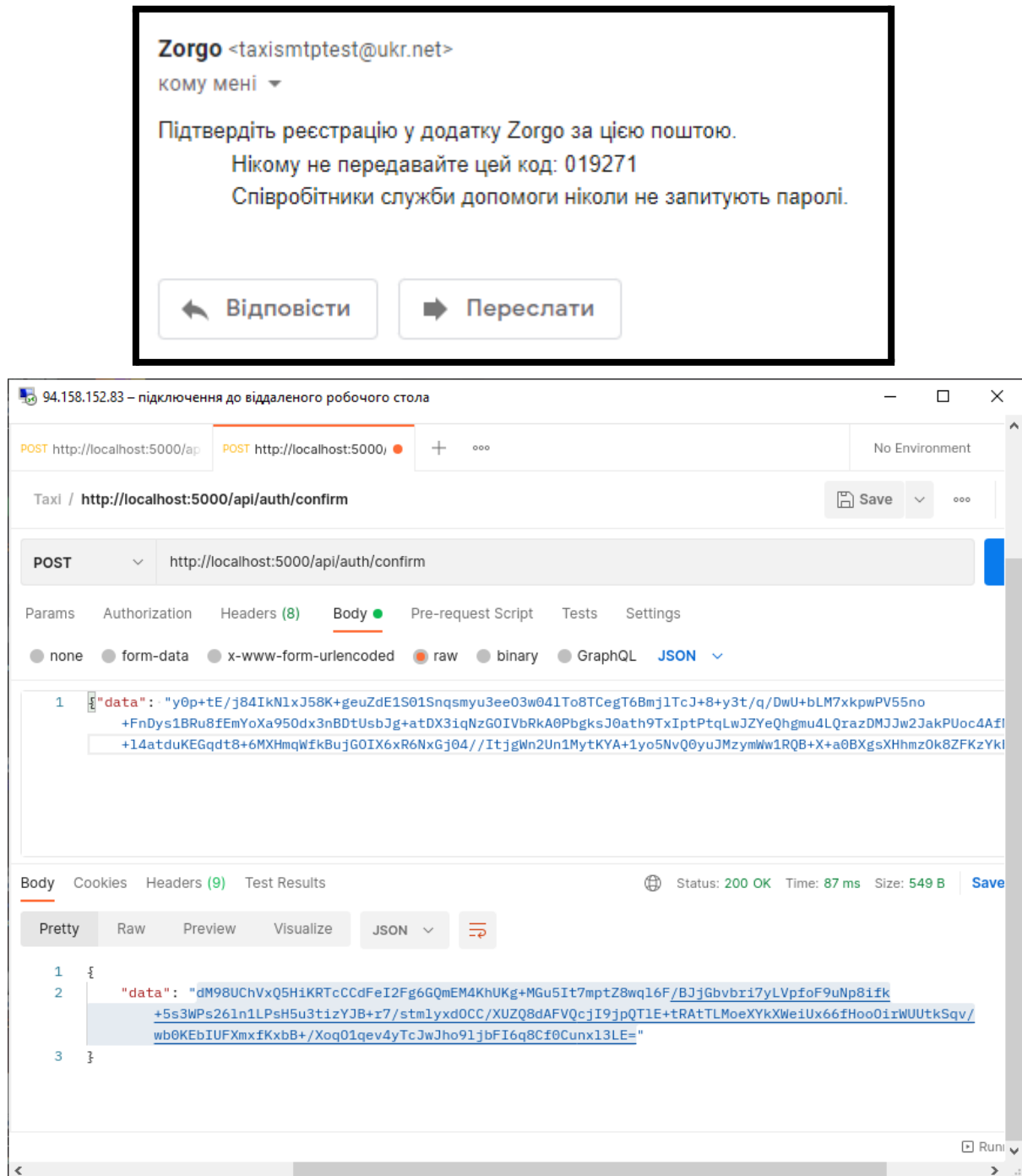


Рисунок 2– Код підтвердження та його перевірка на сервері

У підсумку, всі функції серверної частини, що були регламентовані технічним завданням було реалізовано та випробувано. Поточна версія ПЗ, що реалізує серверну частину, доступна у репозиторії https://bitbucket.org/bohdan_denys/taxi-website/src/master/backend/

ВИСНОВКИ

Проведено огляд інформаційних систем, що враховують відомості про локацію користувачів, виявлено, що суттєве поширення мають системи, які пов'язані із наданням транспортних послуг. Саме для таких систем відомості про локацію мають основне значення. У якості прикладу вибрано для реалізації систему замовлення пасажирських перевезень (таксі).

Визначено програмні засоби, які дозволяють працювати з даними користувача, перевагу надано засобам: MySQL, NodeJS, Express. Дані засоби було вжито при створенні системи.

Спроековано, реалізовано та випробувано систему узгодження взаємодії користувачів з урахуванням геопозиціонування її учасників. Система складена за архітектурою «Progressive Web Application» (PWA) та складається з трьох частин, розміщених за адресами:

- «Mockup» <https://www.figma.com/file/gDr7qqU2ZRd0wt3giXqOnf>,
- «Backend» https://bitbucket.org/bohdan_denys/taxi-website/src/master/backend/
- «Frontend» https://github.com/KristarDen/Taxi_android_app.git.

У систему впроваджено засоби покращення інформаційної безпеки: додаткове шифрування даних, що передаються комунікаційним каналом, за алгоритмом AES-256. Також вжито засобів підтвердження персональних даних користувача, зокрема, номера телефона та електронної пошти. Це дозволяє реалізувати у майбутньому засоби двухфакторної автентифікації, а також ускладнює зловмисникам вживання підроблених (невласних) даних для дискредитації системи.

Систему було випробувано шляхом запуску на фізичному носії та проведення тестових сеансів комунікації. Результати випробування підтвердили загальну працездатність створеної системи.

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. Лященко А. А. Особливості реалізації стандартів доступу до баз геопросторових даних в середовищі універсальних СКБД. // Геоінформаційні технології у територіальному управлінні: матеріали II міжнар. наук.-практ. конф. Одеса: ОРІДУ НАДУ, 2015. с. 17-21 URL: http://www.oridu.odessa.ua/7/7/Book_new_2.pdf
3. 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>
4. 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
5. Ладичук Д.О. Проектування бази геопросторових даних. Навчальний посібник / Ладичук Д.О., Шапоринська Н.М. - Херсон. - 2020. - 128 с. ISBN: 978-966-289-344-1
6. Олександр Смичников. 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> (дата звернення 30.05.2022)
7. 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.
8. 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
9. 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
10. Гоменюк С.І. Математичне моделювання геометричних об'єктів у паралельних комп'ютерних системах. Монографія / Гоменюк С.І., Чопоров С.В., Аль-Атамнех Б.Г. - Херсон. - 2019. - 112 с. ISBN: 978-966-916-714-9
11. Булгакова О.С. Методи та системи штучного інтелекту: теорія та практика. Навчальний посібник / Булгакова О.С., Зосімов В.В., Поздєєв В.О. - Херсон. - 2020. - 356 с. ISBN: 978-966-289-364-9
12. 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
13. 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
14. Чемакіна О.В. Дизайн системи візуальної інформації. Навчальний посібник / Чемакіна О.В., Рубцов А.П., Свірко В.О. - Херсон. - 2019. - 200 с. ISBN: 978-966-289-216-1
15. Шеховцов А.В. Комп'ютерні технології для дизайнерів. Навчальний посібник (стереотипне видання) / Шеховцов А.В., Полетаєва Г.Н., Крючковський Д.О., Бараненко Р.В. - Херсон. - 2019. - 318 с. ISBN: 978-966-2393-08-8
16. 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