

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СПІЛЬНОГО
ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ
SOFTWARE FOR SHARED SMALL SPACE MANAGEMENT

Виконав(ла): студент(ка) 4 курсу, групи КНТ-132

Спеціальності 121 Інженерія програмного

(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

КОЧЕВА М.Г.

(ПРИЗВИЩЕ та ініціали)

Керівник ЛЬОВКІН В.М.

(ПРИЗВИЩЕ та ініціали)

Рецензент ЗЕЛІК О.В.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Факультет КНТ
 Кафедра програмних засобів
 Ступінь вищої освіти бакалавр
 Спеціальність 121 Інженерія програмного забезпечення
(код і найменування)
 Освітня програма (спеціалізація) Інженерія програмного забезпечення
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
 “ ” 2026 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

КОЧЕВОЇ Мілени Геннадіївни
(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення спільного використання невеликих приміщень. Software for Shared Small Space Management

керівник проєкту (роботи) к.т.н., доцент, ЛЬОВКІН Валерій Миколайович,
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 07 ” квітня 2026 року № 139

2. Строк подання студентом проєкту (роботи) 10 червня 2026 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Проєктування програми. 3. Розробка програми. 4. Експлуатація та тестування програми.

5. Перелік графічного матеріалу (з з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	ЛЬОВКІН В.М., доцент		
Нормоконтроль	БСЛОВА А.В., асистент		

7. Дата видачі завдання « 20 » квітня 2026 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка архітектури програми.	2 тиждень	Розділ 2
5	Розробка програми.	3-4 тижні	Розділ 3, 4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

(підпис) Мілена КОЧЕВА
(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Валерій ЛЬОВКІН
(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 94 с., 3 табл., 27 рис., 3 дод., 10 джерел.

НЕВЕЛИКЕ ПРИМІЩЕННЯ, СПІЛЬНЕ ВИКОРИСТАННЯ, ВІДГУК, ВЕБЗАСТОСУНОК, БАЗА ДАНИХ.

Об'єкт роботи – процес розробки програмного забезпечення спільного використання невеликих приміщень. Предмет роботи – програмне забезпечення спільного використання невеликих приміщень.

Мета роботи – зменшення витрат на оренду невеликих приміщень для зберігання речей шляхом розділення цих витрат між різними користувачами на основі розробки програмного забезпечення спільного використання невеликих приміщень.

Матеріали, методи та технічні засоби: мова Python, фреймворк Django як основні технології веброзробки, система PostgreSQL для забезпечення підтримки сховища даних.

Результати. Вибір основних можливостей та особливостей програми було виконано через порівняння з аналогами, проєктування і розробку було забезпечено у відповідності з цими вимогами.

Висновки. Розроблена програма може використовуватись як власниками невеликих приміщень, так і потенційними або фактичними орендарями для того, щоб розподілити наявний простір у невеликих приміщеннях для оренди різних користувачів або запропонувати для спільного використання весь наявний простір невеликого приміщення, керувати запитами зацікавлених осіб у такій оренді, шукати варіанти, які влаштовують користувачів.

Галузь використання. Оренда невеликих приміщень для спільного використання, зокрема зберігання речей, від індивідуальних власників індивідуальними орендарями.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 94 pages, 3 tables, 27 figures, 3 appendixes, 10 sources.

SMALL SPACE, SHARING, FEEDBACK, WEB APPLICATION, DATABASE.

The object of the work is the process of developing software for sharing small spaces. The subject of the work is software for sharing small spaces.

The purpose of the work is to reduce the cost of renting small spaces for storing things by dividing these costs between different users based on the development of software for sharing small spaces.

Materials, methods and technical means: Python language, Django framework as the main web development technologies, PostgreSQL system to provide data storage support.

Results. The selection of the main capabilities and features of the program was carried out through comparison with analogues, design and development were provided in accordance with these requirements.

Conclusions. The developed software can be used by both owners of small spaces and potential or actual tenants in order to distribute the available space in small premises for rent to different users or to offer for shared use all the available space of a small spaces, manage requests from interested parties in such a rent, and search for options that suit users.

Field of use. Rent of small spaces for shared use, including storage of things, from individual owners to individual tenants.

ЗМІСТ

	С.
Перелік скорочень та умовних познач.....	8
Вступ.....	9
1 Аналіз предметної області.....	10
1.1 Огляд програмних засобів спільного використання невеликих приміщень.....	10
1.2 Порівняльний аналіз програмного забезпечення спільного використання невеликих приміщень.....	14
1.3 Висновки за розділом 1	15
2 Проєктування програми.....	16
2.1 Вибір засобів проєктування і розробки	16
2.2 Моделювання роботи з програмою спільного використання невеликих приміщень.....	17
2.3 Визначення структури бази даних програми спільного використання невеликих приміщень.....	22
2.4 Висновки за розділом 2	29
3 Розробка програми	31
3.1 Послідовність роботи з програмою спільного використання невеликих приміщень.....	31
3.2 Рішення з реалізації програми спільного використання невеликих приміщень.....	33
3.3 Висновки за розділом 3	46
4 Експлуатація та тестування програми	47
4.1 Призначення програми	47
4.2 Умови виконання програми	47
4.3 Експлуатація програми спільного використання невеликих приміщень.....	47
4.4 Тестування програми спільного використання невеликих приміщень.....	52

4.5 Висновки за розділом 4	55
Висновки	57
Перелік джерел посилення	59
Додаток А Технічне завдання	60
Додаток Б Текст програми	66
Додаток В Слайди презентації.....	87

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

БД – база даних;

ПЗ – програмне забезпечення.

ВСТУП

Люди нерідко потребують простору, де вони могли б зберігати речі, таким чином зменшуючи обсяг необхідного простору вдома. Для такого зберігання можуть бути потрібні невеликі приміщення або навіть частини кімнат, гаражів та подібних споруд. Такі приміщення можуть бути навіть розділені на використовуватися різними орендарями в підсумку. Власник може зберігати в частині такого приміщення власні речі, а іншу пропонувати для оренди іншим користувачам. Така логіка є доцільною в умовах, коли оренда такого невеликого приміщення може бути доволі високою і часом дорівнювати вартості оренди повноцінного житла. Щоб знизити цю вартість, спільне використання може бути підходящою моделлю для потенційно великої кількості користувачів. Реалізація такої концепції має актуальність та має бути забезпечена в цій роботі. Для цього необхідно розробити програмне забезпечення (ПЗ) спільного використання невеликих приміщень.

Враховуючи специфіку завдання та описану актуальність, метою роботи є зменшення витрат на оренду невеликих приміщень для зберігання речей шляхом розділення цих витрат між різними користувачами на основі розробки програмного забезпечення спільного використання невеликих приміщень. Набір завдань за цією метою:

- аналіз проблеми розробки ПЗ спільного використання невеликих приміщень;
- проєктування ПЗ спільного використання невеликих приміщень;
- розробка ПЗ спільного використання невеликих приміщень.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд програмних засобів спільного використання невеликих приміщень

Neighbor (рис. 1.1) є застосунком, який пропонує можливості оренди невеликих приміщень для збереження речей, паркування автомобілів [1].

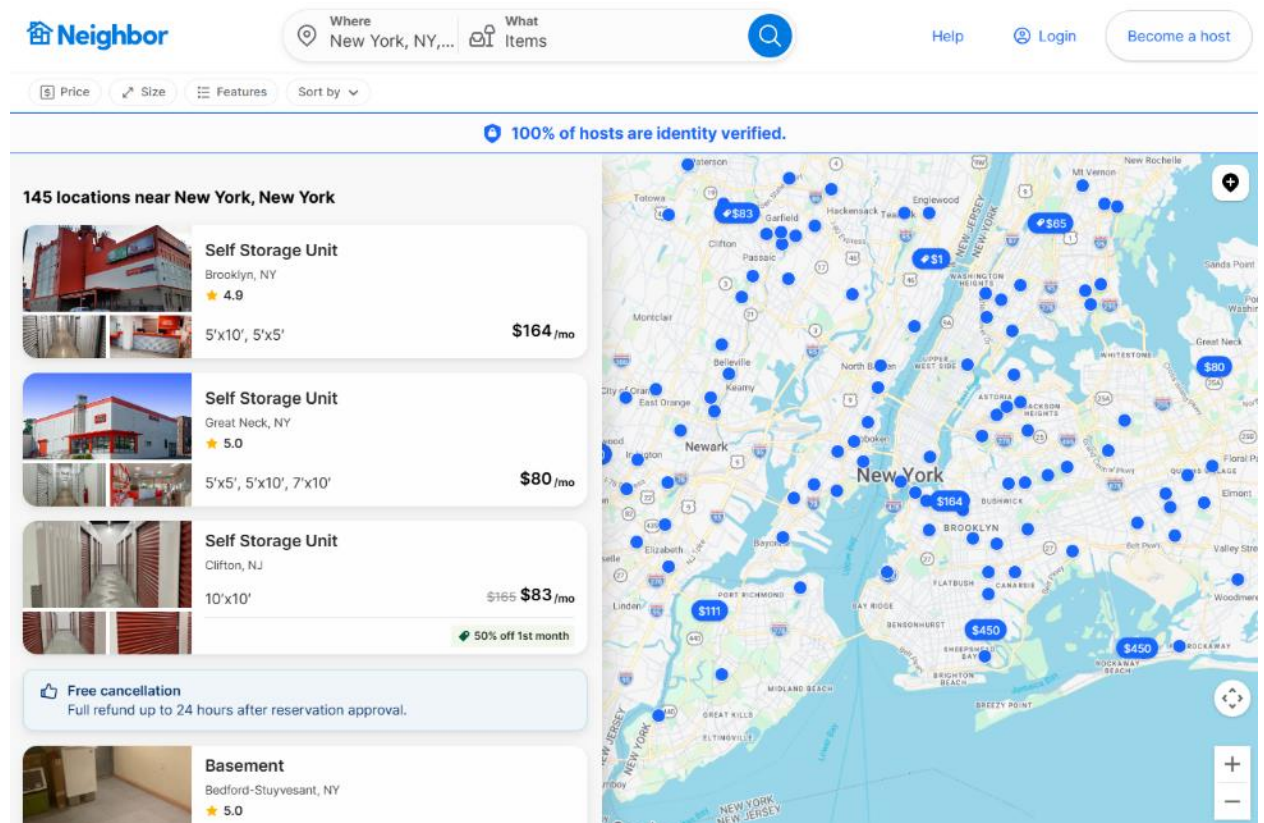


Рисунок 1.1 – Інтерфейс програми Neighbor під час пошуку

Neighbor доступний і як вебзастосунок, і як застосунок для мобільних операційних систем, включаючи Android та iOS. Neighbor надає можливості створення окремих зон для організації збереження в приміщеннях, проте кожна така зона повністю має бути відокремлена. З іншого боку частина власників невеликих приміщень можуть не мати можливості, бажання відділяти такий простір. Цей підхід може вимагати меншої сплати, але може бути цікавим для збереження речей певним орендарям. Наприклад, якщо орендар хоче використовувати приміщення для збереження велосипеда, то

його може влаштувати співіснування з власником або іншим орендарем на одній території. При цьому загальне приміщення буде відділене від іншої території, буде захищеним від всіх інших людей. Такий варіант може бути підходящим. Але він не забезпечується Neighbor.

У Neighbor немає розділення невеликого приміщення на окремі ідентифіковані складові (рис. 1.2). Тобто якщо доступно багато комірок, то оренда відбувається не за певною коміркою, а комірки певного типу. Самі відгуки користувачів стосуються приміщення загалом, а не окремих комірок.

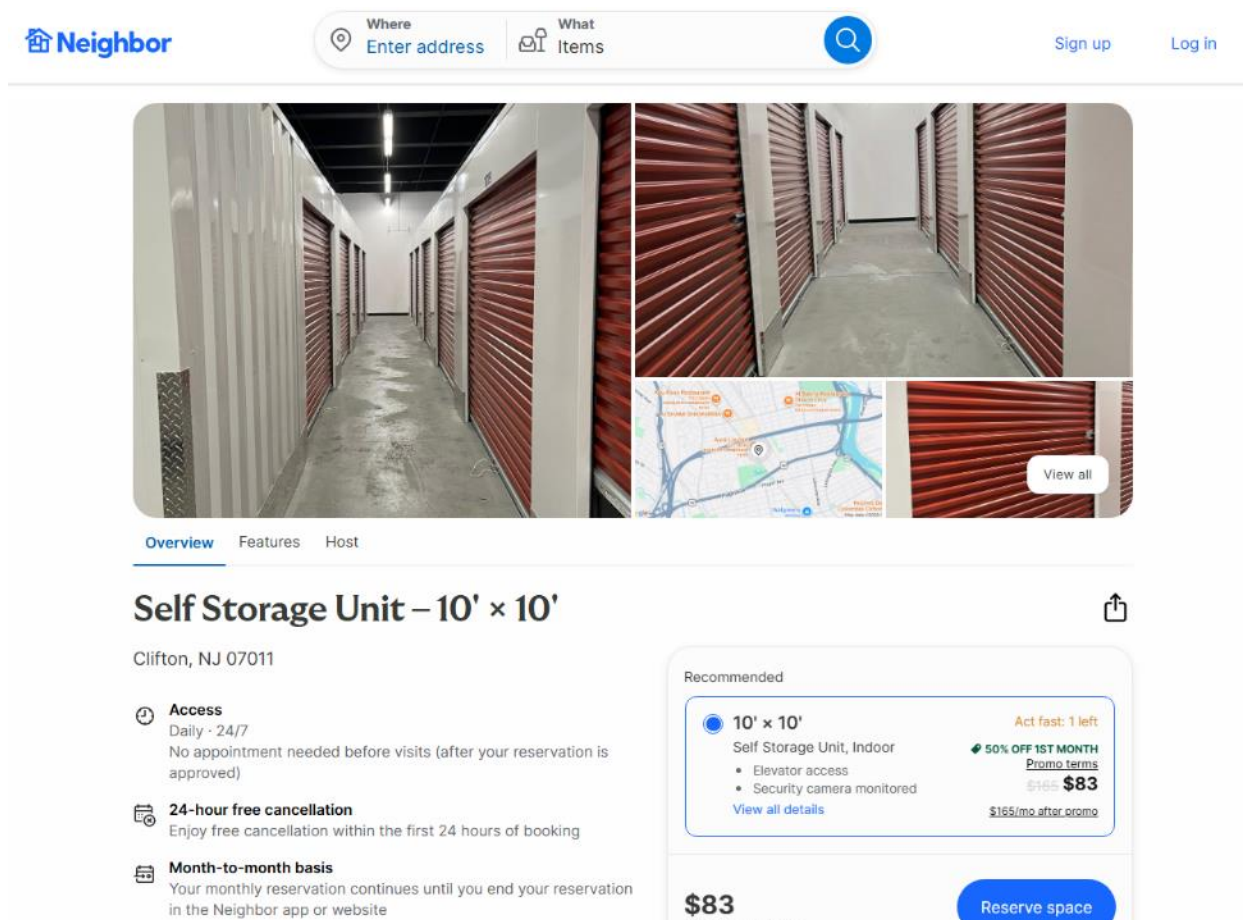


Рисунок 1.2 – Інтерфейс програми Neighbor під час відображення сховища

Shared Storage (рис. 1.3) є подібним застосунком, який при тому доступний для використання тільки на мобільних пристроях, включаючи операційні системи Android і iOS [2].

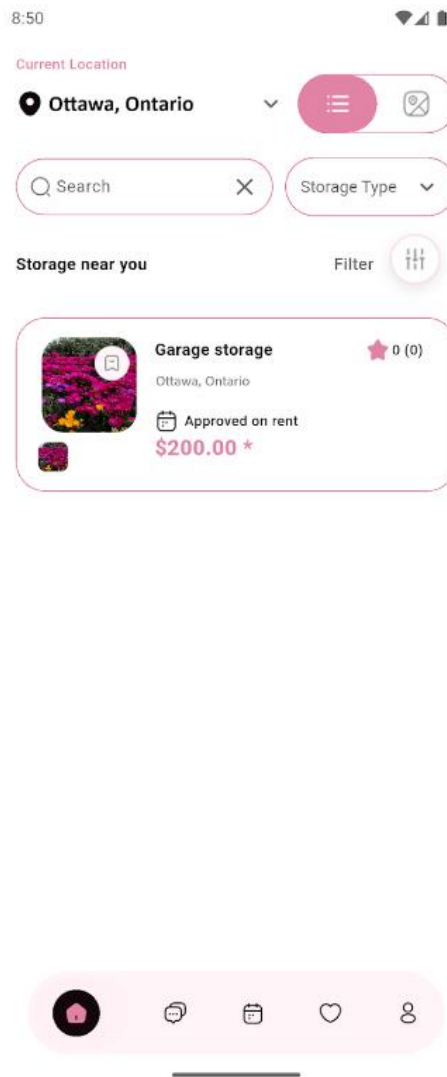


Рисунок 1.3 – Інтерфейс програми Shared Storage під час пошуку [3]

У Shared Storage (рис. 1.4) немає можливості виділення окремих комірок в межах певного приміщення, на відміну від попередньої програми. Доступна можливість оцінювання досвіду оренди таких приміщень, доступна можливість поставити запитання за приміщенням. Відгуки відповідно доступні тільки за приміщенням загалом.

У Shared Storage, як і в Neighbor, оренда приміщення здійснюється на строк у місяць. Відповідно встановити гнучкі тарифи власник не може, аналогічно вони не доступні і орендарям.

**Measurements : (Feet)**

- Length : 10
- Width : 10
- Height : 20

Surface Type

- Concrete

Insulated

- Yes

Disclosure

- No hazardous materials flammable liquids or food storage allowed

Storage Type

- Self-Storage

Access Type

- Restricted

\$200.00 per month

1 month on rent

[View Breakdown](#)

Chat

Рисунок 1.4 – Інтерфейс програми Shared Storage під час відображення сховища [3]

1.2 Порівняльний аналіз програмного забезпечення спільного використання невеликих приміщень

Основними аналогами, які здатні забезпечувати першочерговий набір функцій оренди невеликих приміщень, для ПЗ, що розробляється, є застосунки Neighbor і Shared Storage (табл. 1.1).

Таблиця 1.1 – Порівняння ПЗ спільного використання невеликих приміщень

Критерій	Neighbor	Shared Storage	Розробка
Оренда невеликих приміщень	Так	Так	Так
Спільне використання невеликих приміщень	Однакова модель для всіх складових	Ні	Індивідуальна модель для всіх складових
Базова одиниця періоду оренди невеликих приміщень	Місяць	Місяць	День, місяць, рік
Створення відгуків за варіантом використання невеликого приміщення	Ні	Ні	Так

Детальне порівняння Neighbor і Shared Storage з аналогами дозволило виявити, що Neighbor дозволяє надавати можливості спільного використання невеликих приміщень, формуючи певні, наприклад, комірки для збереження, проте така модель базується на типовості для кожного такого варіанту. Тобто визначати особливості оренди, а далі і у відгуках досвіду за конкретними варіантами спільного використання неможливо. Себто якщо одна комірка має поганий стан, а інша така саме за параметрами таких проблем не має, то

з'ясувати, яка з них в якому стані, було неможливо. Це запропоновано вирішити у розробці ПЗ спільного використання невеликих приміщень. Також базовою одиницею періоду оренди невеликих приміщень у аналогів є місяць. Розробка має запропонувати менший період.

1.3 Висновки за розділом 1

Аналіз проблеми розробки ПЗ спільного використання невеликих приміщень забезпечено в даному розділі, в підсумку чого було спершу виділено аналоги, що включають застосунки Neighbor і Shared Storage, а далі порівняно ці аналоги з розробкою, виділено особливості розробки, які включають визначення конкретних особливостей варіантів спільного використання в межах одного невеликого приміщення, врахування цього в самих запитуваннях оренди, у відгуках, а також визначення можливості оренди на період менше 1 місяця.

2 ПРОЄКТУВАННЯ ПРОГРАМИ

2.1 Вибір засобів проєктування і розробки

Для пришвидшення розробки ПЗ спільного використання невеликих приміщень вирішено було використовувати мову Python, оскільки вона і читабельна, і дозволяє писати більш короткий код. При цьому фреймворки для веброзробки, серед найпопулярніших з яких є Django [4]-[5] і Flask [6]-[7], були порівняні окремо (табл. 2.1). А обрано було Django, оскільки цей фреймворк має також менший час розробки рішень на його основі, більш придатний для великих застосунків, яким потенційно є даний, має вбудовану систему аутентифікації та роботи з даними.

Таблиця 2.1 – Порівняння фреймворків для розробки ПЗ спільного використання невеликих приміщень

Критерій	Django	Flask
Час розробки	Менший за рахунок вбудованих засобів	Більший
Сценарії використання	Великі застосунки, соціальні платформи	Малі застосунки, створення прототипів
Система аутентифікації	Вбудована	Має бути вибрана і встановлена
Робота з даними	Вбудована система адміністрування, об'єктно-реляційне відображення для бази даних	Засоби мають бути вибрані і встановлені

2.2 Моделювання роботи з програмою спільного використання невеликих приміщень

Усі користувачі, які працюють з ПЗ спільного використання невеликих приміщень, розподіляються тільки на 2 ролі. Перша – це неавторизований користувач, тобто користувач, який не має або не використовує свій обліковий запис для роботи з програмою. Друга – авторизований користувач, який відповідно має і використовує свій обліковий запис для роботи з програмою, при цьому він може взаємодіяти з програмою в такій ролі і як власник невеликого приміщення, і як орендар. Тобто будь-який користувач, який має обліковий запис в програмі, може як орендувати невеликі приміщення для спільного використання, так і виставляти свої приміщення для спільного використання іншими користувачами.

Неавторизований користувач ПЗ спільного використання невеликих приміщень має такий основний потік сценаріїв:

- перегляд оголошення про спільне використання невеликого приміщення;
- перегляд сторінки власника невеликих приміщень;
- перегляд сторінки орендаря спільного використання невеликих приміщень;
- перегляд оголошень про спільне використання невеликих приміщень власника;
- перегляд відгуків про досвід спільного використання за невеликим приміщенням;
- перегляд відгуків про досвід спільного використання за власником невеликих приміщень;
- перегляд відгуків про досвід взаємодії за спільним використанням за орендарем;
- пошук невеликих приміщень для спільного використання;
- авторизація облікового запису (рис. 2.1).



Рисунок 2.1 – Діаграма варіантів використання ПЗ спільного використання невеликих приміщень неавторизованим користувачем

Для неавторизованого користувача ПЗ спільного використання невеликих приміщень передбачено тільки один альтернативний сценарій – реєстрація облікового запису, який використовується тоді, коли користувач не має взагалі або не має доступу до облікового запису.

Авторизований користувач ПЗ спільного використання невеликих приміщень має такий основний потік сценаріїв:

- додавання оголошення про спільне використання невеликого приміщення;
- перегляд власних оголошень про спільне використання невеликих приміщень;
- перегляд невеликих приміщень у використанні користувача;
- додавання варіанту спільного використання за невеликим приміщенням;

- нотування оголошення про спільне використання невеликого приміщення;
- перегляд занотованих оголошень про спільне використання невеликих приміщень;
- запитування спільного використання невеликого приміщення;
- підтвердження запитування спільного використання невеликого приміщення;
- заповнення користувацьких даних;
- створення відгуку про досвід спільного використання невеликого приміщення;
- створення відгуку про досвід взаємодії з орендарем спільного використання невеликого приміщення;
- перегляд запитувань спільного використання за невеликим приміщенням;
- перегляд запитувань спільного використання за всіма невеликими приміщеннями власника;
- перегляд підтверджених запитувань спільного використання за всіма невеликими приміщеннями власника;
- перегляд скасованих запитувань спільного використання за всіма невеликими приміщеннями власника;
- перегляд всіх запитувань спільного використання невеликих приміщень за орендарем;
- відправлення повідомлення за невеликим приміщенням для спільного використання;
- перегляд повідомлень за невеликим приміщенням для спільного використання у орендаря;
- перегляд невеликих приміщень для спільного використання, за якими здійснювалась переписка;
- перегляд переписок за невеликим приміщенням для спільного використання у власника;

– додавання періоду доступності невеликого приміщення для спільного використання (рис. 2.2).

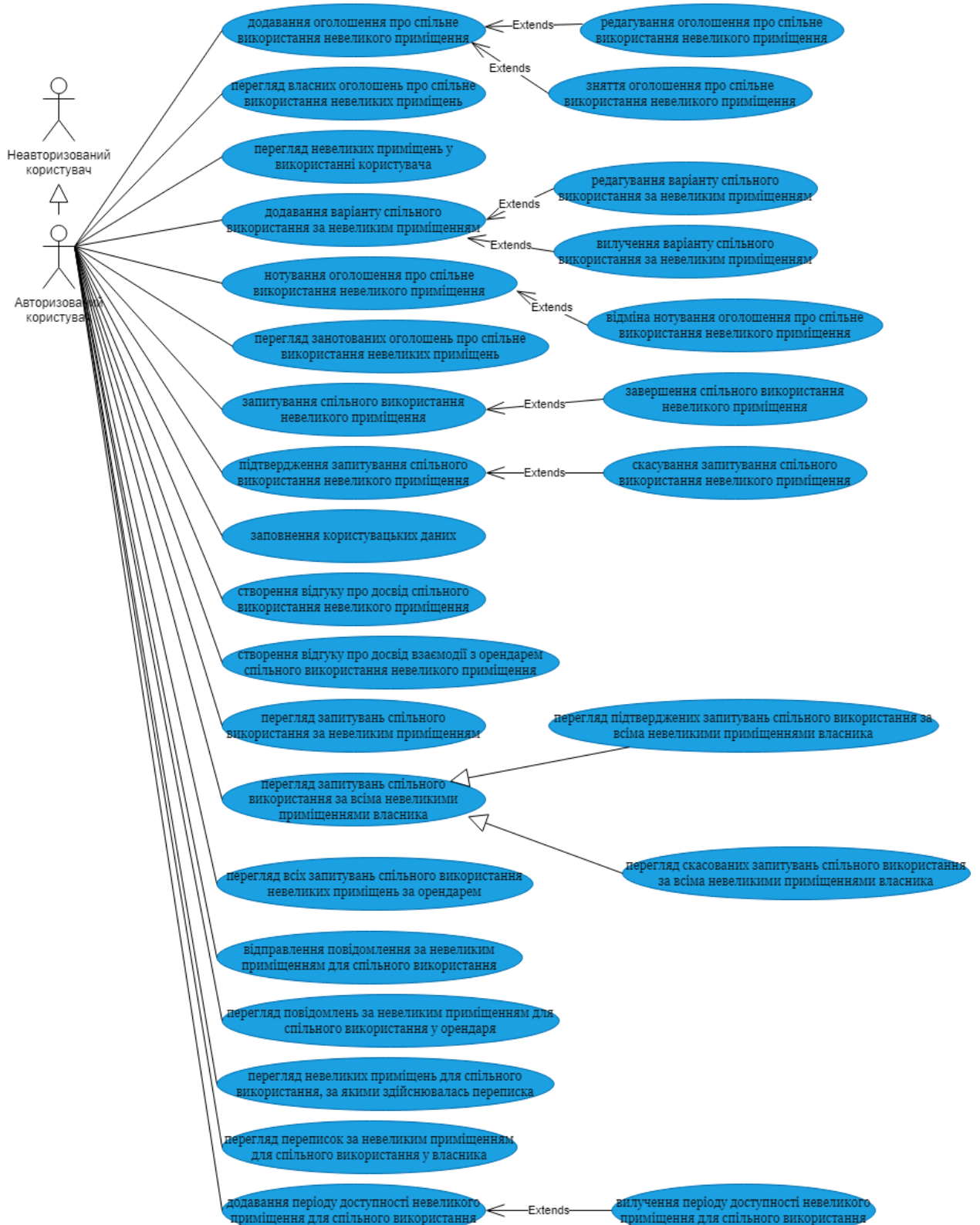


Рисунок 2.2 – Діаграма варіантів використання ПЗ спільного використання невеликих приміщень авторизованим користувачем

Альтернативні сценарії авторизованого користувача ПЗ спільного використання невеликих приміщень:

- редагування оголошення про спільне використання невеликого приміщення, який є альтернативним сценарієм для додавання оголошення про спільне використання невеликого приміщення, коли таке оголошення про спільне використання невеликого приміщення вже додано, але має бути змінено для відповідності поточним потребам;

- зняття оголошення про спільне використання невеликого приміщення, який є альтернативним сценарієм для додавання оголошення про спільне використання невеликого приміщення, коли таке оголошення про спільне використання невеликого приміщення вже додано, але в поточний момент воно не має використовуватися, тому що з тих чи інших причин власник не бажає надавати його для спільного використання;

- редагування варіанту спільного використання за невеликим приміщенням, який є альтернативним сценарієм для додавання варіанту спільного використання за невеликим приміщенням, коли такий варіант спільного використання за невеликим приміщенням вже доданий, але має бути змінений для відповідності поточним потребам;

- вилучення варіанту спільного використання за невеликим приміщенням, який є альтернативним сценарієм для додавання варіанту спільного використання за невеликим приміщенням, коли такий варіант спільного використання за невеликим приміщенням вже доданий, але це було зроблено помилково або більше не має потреби колись його використовувати в майбутньому;

- відміна нотування оголошення про спільне використання невеликого приміщення, який є альтернативним сценарієм для нотування оголошення про спільне використання невеликого приміщення, коли таке нотування нотування оголошення про спільне використання невеликого приміщення вже виконано цим користувачем, але більше йому не потрібне;

– скасування запитування спільного використання невеликого приміщення, який є альтернативним сценарієм для підтвердження запитування спільного використання невеликого приміщення, коли власник не вважає можливим надати своє невелике приміщення для спільного використання тим орендарем, який про це запитав;

– вилучення періоду доступності невеликого приміщення для спільного використання, який є альтернативним сценарієм для додавання періоду доступності невеликого приміщення для спільного використання, коли такий період доступності невеликого приміщення вже був визначений, але більше не має застосовуватися;

– завершення спільного використання невеликого приміщення, який є альтернативним сценарієм для запитування спільного використання невеликого приміщення, коли таке спільне використання невеликого приміщення вже розпочалось, але має бути завершене раніше.

2.3 Визначення структури бази даних програми спільного використання невеликих приміщень

База даних (БД) ПЗ спільного використання невеликих приміщень організована під керуванням системи PostgreSQL. Структура ж БД описана нижче.

Таблиця Spacescountry БД представляє країни, в яких можуть знаходитись в програмі невеликі приміщення для спільного використання, складається з полів:

– id – ідентифікатор країни, в якій можуть знаходитись в програмі невеликі приміщення для спільного використання;

– spacescountryname – назва країни, в якій можуть знаходитись в програмі невеликі приміщення для спільного використання.

Таблиця Spacescity БД представляє міста, в яких можуть знаходитись в програмі невеликі приміщення для спільного використання, складається з полів:

- id – ідентифікатор міста, в якому можуть знаходитись в програмі невеликі приміщення для спільного використання;
- spacescountryid – країна, в якій знаходиться місто, в якому можуть знаходитись в програмі невеликі приміщення для спільного використання (зовнішній ключ до таблиці Spacescountry);
- spacescityname – назва міста, в якому можуть знаходитись в програмі невеликі приміщення для спільного використання.

Таблиця Smallspacetype БД представляє типи невеликих приміщень для спільного використання, складається з полів:

- id – ідентифікатор типу невеликого приміщення для спільного використання;
- smallspacetype name – назва типу невеликого приміщення для спільного використання.

Таблиця Smallspace БД представляє оголошення про спільне використання невеликого приміщення, складається з полів:

- id – ідентифікатор оголошення про спільне використання невеликого приміщення;
- smallspacetypeid – тип невеликого приміщення для спільного використання, визначеного в оголошенні (зовнішній ключ до таблиці Smallspacetype);
- spacescityid – місто, в якому знаходиться невелике приміщення для спільного використання з оголошення (зовнішній ключ до таблиці Spacescity);
- smallspaceownerid – власник невеликого приміщення для спільного використання, визначеного в оголошенні (зовнішній ключ до таблиці Spaceshareuser);
- smallspacename – назва оголошення про спільне використання невеликого приміщення;

– smallspaceadvertisement – опис з оголошення про спільне використання невеликого приміщення;

– buildingtype – тип будівлі, в якій знаходиться це невелике приміщення для спільного використання;

– smallspacereviewmark – усереднена оцінка відгуків про спільне використання цього невеликого приміщення;

– smallspacearea – сумарна площа невеликого приміщення, яке пропонується для спільного використання;

– smallspacerentrequirements – вимоги до орендарів для спільного використання пропонованого невеликого приміщення.

Таблиця Smallspacephoto БД представляє фотографії невеликих приміщень для спільного використання, складається з полів:

– id – ідентифікатор фотографії невеликого приміщення для спільного використання;

– smallspaceid – невелике приміщення для спільного використання, представлене на фотографії (зовнішній ключ до таблиці Smallspace);

– smallspacephotodata – дані фотографії невеликого приміщення для спільного використання.

Таблиця Smallspaceshare БД представляє варіанти спільного використання за невеликим приміщенням (фактично це певна частинка, наприклад, кімрка у невеликому приміщенні, яка представляється для окремої оренди), складається з полів:

– id – ідентифікатор варіанту спільного використання за невеликим приміщенням;

– smallspaceid – невелике приміщення, за яким надається цей варіант спільного використання (зовнішній ключ до таблиці Smallspace);

– smallspacesharestate – стан варіанту спільного використання за невеликим приміщенням, який необхідний для того, щоб визначити недоступність такого варіанту на певний час, але не видаляти його при цьому повністю;

- smallspacesharetype – тип спільного використання невеликого приміщення (приміщення повністю, з господарем, з господарем та орендарями, з орендарями);
- smallspacesharename – назва цього варіанту спільного використання за невеликим приміщенням;
- smallspaceshareadvertisement – опис цього варіанту спільного використання за невеликим приміщенням;
- smallspacesharereviewmark – усереднена оцінка з відгуків про спільне використання цього варіанту невеликого приміщення;
- smallspacesharevolume – обсяг в кубічних метрах, доступний за цим варіантом спільного використання цього невеликого приміщення;
- smallspaceshareheight – висота простору, доступного за цим варіантом спільного використання цього невеликого приміщення;
- smallspacesharewidth – ширина простору, доступного за цим варіантом спільного використання цього невеликого приміщення;
- smallspacesharelength – довжина простору, доступного за цим варіантом спільного використання цього невеликого приміщення;
- smallspaceaccesstype – тип доступу для спільного використання невеликого приміщення (окремі входи, спільний вхід з окремими кімнатами, що закриваються, спільний вхід з окремими кімнатами, що не закриваються, спільне приміщення);
- smallspacesharedayrentprice – вартість оренди цього варіанту спільного використання невеликого приміщення на добу (базова вартість в євро);
- smallspacesharemonthrentprice – вартість оренди цього варіанту спільного використання невеликого приміщення на місяць (базова вартість в євро);
- smallspaceshareyearrentprice – вартість оренди цього варіанту спільного використання невеликого приміщення на рік (базова вартість в євро).

Таблиця Smallspaceshareaccess БД представляє періоди доступності невеликого приміщення для спільного використання, складається з полів:

- id – ідентифікатор періоду доступності невеликого приміщення для спільного використання;
- smallspaceshareid – оголошення про спільне використання невеликого приміщення, до якого відноситься цей визначений період доступності невеликого приміщення (зовнішній ключ до таблиці Smallspaceshare);
- smallspacesharestartdate – початок періоду доступності невеликого приміщення для спільного використання;
- smallspaceshareenddate – завершення періоду доступності невеликого приміщення для спільного використання.

Таблиця Smallspacesharephoto БД представляє фотографії варіантів спільного використання за невеликим приміщенням, складається з полів:

- id – ідентифікатор фотографії варіанта спільного використання за невеликим приміщенням;
- smallspaceshareid – варіант спільного використання за невеликим приміщенням, який показаний на цій фотографії (зовнішній ключ до таблиці Smallspaceshare);
- smallspacesharephotodata – дані фотографії варіанта спільного використання за невеликим приміщенням.

Таблиця Smallspacerequest БД представляє запитування спільного використання невеликого приміщення, складається з полів:

- id – ідентифікатор запитування спільного використання невеликого приміщення;
- smallspaceshareid – варіант спільного використання невеликого приміщення, якого стосується це запитування (зовнішній ключ до таблиці Smallspaceshare);
- spaceshareuserid – користувач, який запитав спільне використання цього невеликого приміщення (зовнішній ключ до таблиці Spaceshareuser);

– smallspacerequeststate – стан запитування спільного використання невеликого приміщення;

– smallspacerequestdate – час запитування спільного використання невеликого приміщення;

– smallspacerequestnote – нотатка до запитування спільного використання невеликого приміщення;

– smallspacesharestartdate – час початку спільного використання невеликого приміщення за цим запитом;

– smallspaceshareenddate – час закінчення спільного використання невеликого приміщення за цим запитом.

Таблиця Smallspacenote БД представляє нотування оголошення про спільне використання невеликого приміщення, складається з полів:

– id – ідентифікатор нотування оголошення про спільне використання невеликого приміщення;

– smallspaceid – оголошення про спільне використання невеликого приміщення, до якого зроблено це нотування (зовнішній ключ до таблиці Smallspace);

– spaceshareuserid – користувач, який занотував це оголошення про спільне використання невеликого приміщення (зовнішній ключ до таблиці Spaceshareuser);

– smallspacenotetext – текст нотатки до оголошення про спільне використання невеликого приміщення;

– smallspacetypeid – тип невеликого приміщення для спільного використання, до якого віднесено цю збережену нотатку (зовнішній ключ до таблиці Smallspacetype).

Таблиця Spaceshareuser БД представляє користувача, складається з полів:

– id – ідентифікатор користувача;

– spaceshareraccount – обліковий запис цього користувача в Django (зовнішній ключ до таблиці User);

- spacesharerphone – номер телефону користувача;
- spaceshareradvertisement – опис користувача;
- spacesharerrentreviewmark – усереднена оцінка користувача як орендаря спільного використання невеликих приміщень;
- spacesharerphoto – фотографія користувача;
- spacecityid – місто, в якому проживає користувач (зовнішній ключ до таблиці Spacecity).

Таблиця Smallspacesharereview БД представляє відгуки про досвід спільного використання невеликих приміщень, складається з полів:

- id – ідентифікатор відгуку про досвід спільного використання невеликого приміщення;
- smallspacerequestid – запитування спільного використання невеликого приміщення, за яким сформовано цей відгук (зовнішній ключ до таблиці Smallspacerequest);
- smallspacesharereviewdate – час створення відгуку про досвід спільного використання невеликого приміщення;
- smallspacesharereviewtext – текст відгуку про досвід спільного використання невеликого приміщення;
- smallspacesharereviewscore – оцінка відгуку про досвід спільного використання невеликого приміщення;
- smallspacesharereviewside – сторона, яка створила відгук про досвід спільного використання невеликого приміщення: власник невеликого приміщення або орендар.

Таблиця Spacesharemessage БД представляє повідомлення за невеликим приміщенням для спільного використання, складається з полів:

- id – ідентифікатор повідомлення за невеликим приміщенням для спільного використання;
- smallspaceid – оголошення про спільне використання невеликого приміщення, за яким створено це повідомлення;

- spaceshareuserid – користувач, який створив це повідомлення (зовнішній ключ до таблиці Spaceshareuser);
- spacesharemessagedate – час створення повідомлення за невеликим приміщенням для спільного використання;
- spacesharemessagetext – текст повідомлення за невеликим приміщенням для спільного використання (рис. 2.3).

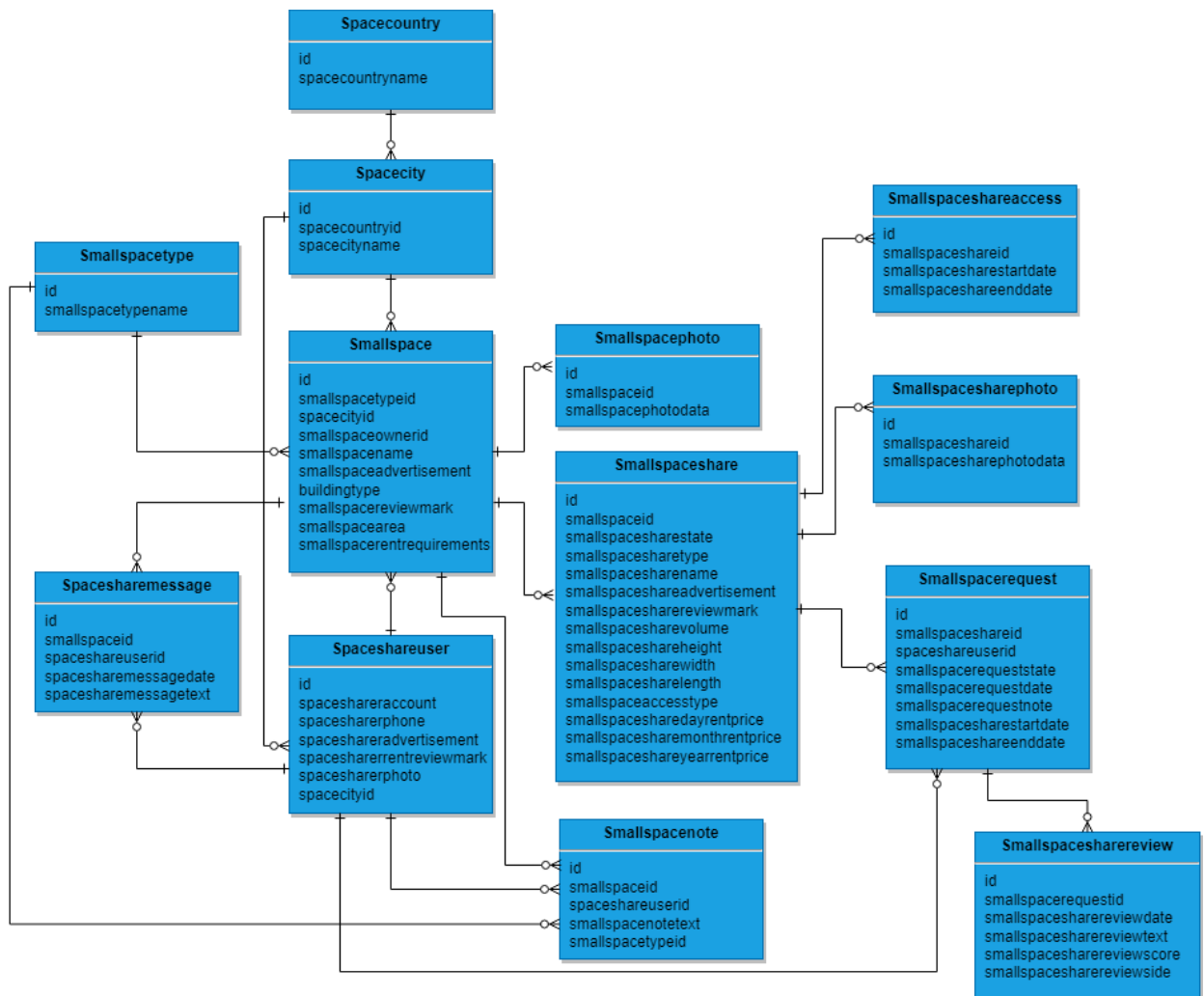


Рисунок 2.3 – Схема БД ПЗ спільного використання невеликих приміщень

2.4 Висновки за розділом 2

Проектування ПЗ спільного використання невеликих приміщень у даному розділі було забезпечено через моделювання роботи користувачів у двох ролях з програмою (неавторизований та авторизований користувач),

визначення структури БД та вибір засобів розробки і його обґрунтування, включаючи мову програмування Python, фреймворк Django, систему PostgreSQL.

3 РОЗРОБКА ПРОГРАМИ

3.1 Послідовність роботи з програмою спільного використання невеликих приміщень

Програма спільного використання невеликих приміщень (рис. 3.1) базується на наданні користувачу функції пошуку невеликих приміщень для спільного використання, яка у випадку подальшої авторизації користувача, якщо вона ще не була виконана раніше, далі розширюється додатковими можливостями для роботи, які згруповані за розділами, що передбачають наступне:

- керування власними оголошеннями про спільне використання невеликого приміщення;
- керування запитуваннями спільного використання невеликого приміщення;
- взаємодію з переписками;
- перегляд невеликих приміщень у використанні користувача;
- керування даними користувача;
- керування занотованими оголошеннями про спільне використання невеликих приміщень.

Користувач може перемикатися між всіма цими розділами, обираючи потрібний, доки не вирішить взагалі завершити роботу з програмою або вийти з облікового запису.

У випадку виходу з облікового запису користувач знову отримує можливість виконання пошуку невеликих приміщень для спільного використання, завершення роботи або авторизації повторно або за іншим обліковим записом.

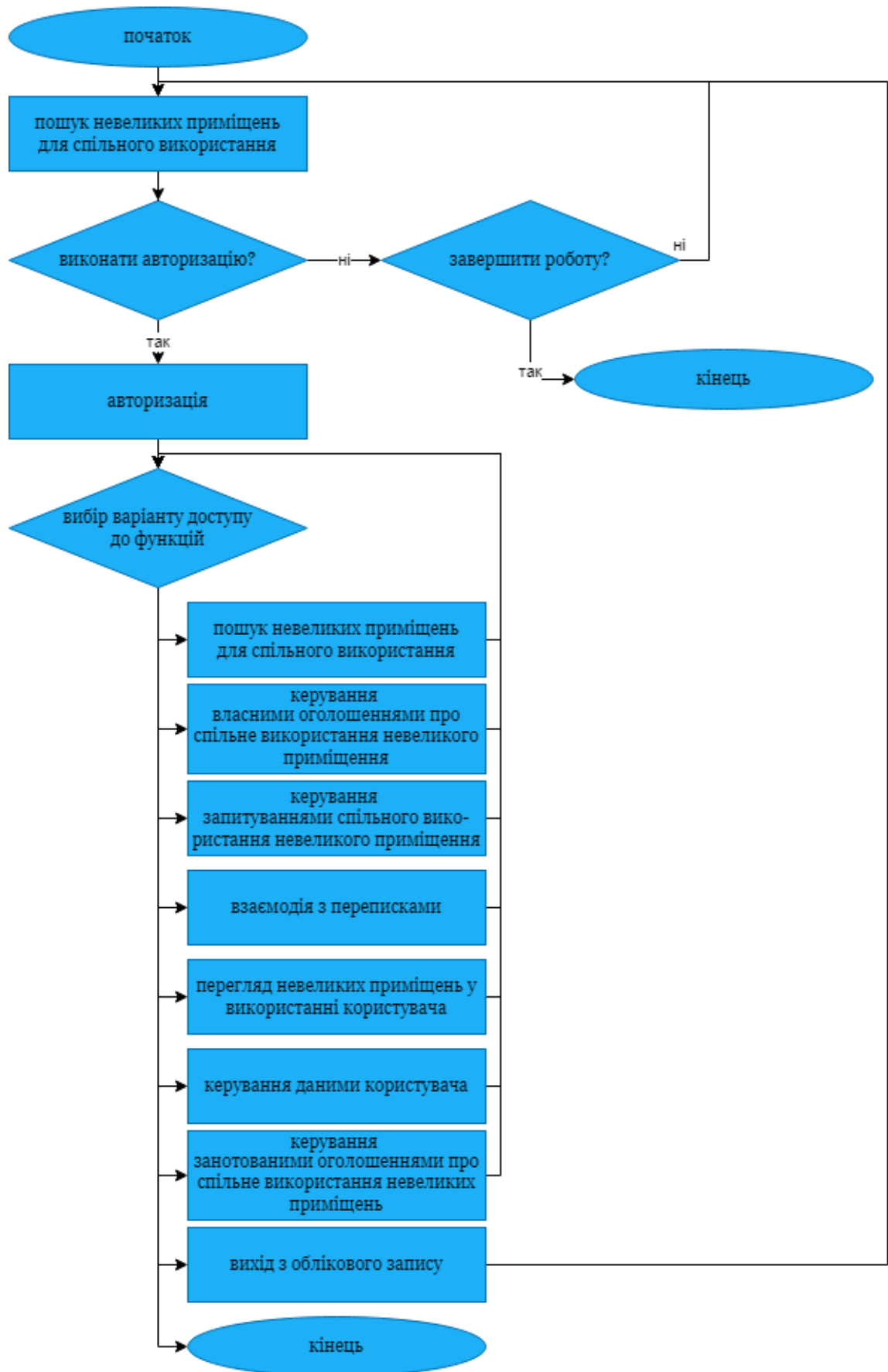


Рисунок 3.1 – Схема функціонування ПЗ спільного використання невеликих приміщень

3.2 Рішення з реалізації програми спільного використання невеликих приміщень

ПЗ спільного використання невеликих приміщень організовано під час реалізації за логікою застосунку за фреймворком Django, який було обрано в розділі 2. Відповідно логіка роботи з даними побудована на класах моделей [8], які і описані нижче.

Клас `Spacecountry` реалізує модель країн, в яких можуть знаходитись в програмі невеликі приміщення для спільного використання, надаючи для цього поле `spacecountryname` типу `CharField` з назвою країни, в якій можуть знаходитись в програмі невеликі приміщення для спільного використання.

Клас `Spacecity` реалізує модель міст, в яких можуть знаходитись в програмі невеликі приміщення для спільного використання.

Клас `Spacecity` має поле `spacecountry` типу `ForeignKey`, що реалізує країну, в якій знаходиться місто, в якому можуть знаходитись в програмі невеликі приміщення для спільного використання, зі зв'язком з моделлю `Spacecountry`.

Клас `Spacecity` має поле `spacecityname` типу `CharField` з назвою міста, в якому можуть знаходитись в програмі невеликі приміщення для спільного використання.

Клас `Smallspacetype` реалізує модель типів невеликих приміщень для спільного використання. Клас має поле `smallspacetype` типу `CharField` з назвою типу невеликого приміщення для спільного використання.

Клас `Smallspace` реалізує модель оголошення про спільне використання невеликого приміщення.

Клас `Smallspace` має поле `smallspacetype` типу `ForeignKey`, що реалізує тип невеликого приміщення для спільного використання, визначеного в оголошенні, зі зв'язком з моделлю `Smallspacetype`.

Клас `Smallspace` має поле `spacecity` типу `ForeignKey`, що реалізує місто, в якому знаходиться невелике приміщення для спільного використання з оголошення, зі зв'язком з моделлю `Spacecity`.

Клас `Smallspace` має поле `smallspaceowner` типу `ForeignKey`, що реалізує власника невеликого приміщення для спільного використання, визначеного в оголошенні, зі зв'язком з моделлю `Spaceshareuser`.

Клас `Smallspace` має поле `smallspacename` типу `CharField` з назвою оголошення про спільне використання невеликого приміщення.

Клас `Smallspace` має поле `smallspaceadvertisement` типу `TextField` з описом з оголошення про спільне використання невеликого приміщення.

Клас `Smallspace` має поле `buildingtype` типу `CharField` з типом будівлі, в якій знаходиться це невелике приміщення для спільного використання.

Клас `Smallspace` має поле `smallspacereviewmark` типу `DecimalField` з усередненою оцінкою відгуків про спільне використання цього невеликого приміщення.

Клас `Smallspace` має поле `smallspacearea` типу `DecimalField` з сумарною площею невеликого приміщення, яке пропонується для спільного використання.

Клас `Smallspace` має поле `smallspacerentrequirements` типу `CharField` з вимогами до орендарів для спільного використання пропонованого невеликого приміщення.

Клас `Smallspacephoto` реалізує модель фотографій невеликих приміщень для спільного використання.

Клас `Smallspacephoto` має поле `smallspace` типу `ForeignKey`, що реалізує невелике приміщення для спільного використання, представлене на фотографії, зі зв'язком з моделлю `Smallspace`.

Клас `Smallspacephoto` має поле `smallspacephotodata` типу `ImageField` з даними фотографії невеликого приміщення для спільного використання.

Клас `Smallspaceshare` реалізує модель варіантів спільного використання за невеликим приміщенням.

Клас `Smallspaceshare` має поле `smallspace` типу `ForeignKey`, що реалізує невелике приміщення, за яким надається цей варіант спільного використання, зі зв'язком з моделлю `Smallspace`.

Клас `Smallspaceshare` має поле `smallspacesharestate` типу `CharField` зі станом варіанту спільного використання за невеликим приміщенням, який необхідний для того, щоб визначити недоступність такого варіанту на певний час, але не видаляти його при цьому повністю.

Клас `Smallspaceshare` має поле `smallspacesharetype` типу `CharField` з типом спільного використання невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspacesharename` типу `CharField` з назвою цього варіанту спільного використання за невеликим приміщенням.

Клас `Smallspaceshare` має поле `smallspaceshareadvertisement` типу `TextField` з описом цього варіанту спільного використання за невеликим приміщенням.

Клас `Smallspaceshare` має поле `smallspacesharereviewmark` типу `DecimalField` з усередненою оцінкою з відгуків про спільне використання цього варіанту невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspacesharevolume` типу `DecimalField` з обсягом в кубічних метрах, доступний за цим варіантом спільного використання цього невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspaceshareheight` типу `DecimalField` з висотою простору, доступного за цим варіантом спільного використання цього невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspacesharewidth` типу `DecimalField` з шириною простору, доступного за цим варіантом спільного використання цього невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspacesharelength` типу `DecimalField` з довжиною простору, доступного за цим варіантом спільного використання цього невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspaceaccesstype` типу `CharField` з типом доступу для спільного використання невеликого приміщення.

Клас `Smallspaceshare` має поле `smallspacesharedayrentprice` типу `DecimalField` з вартістю оренди цього варіанту спільного використання невеликого приміщення на добу (базова вартість в євро).

Клас `Smallspaceshare` має поле `smallspacesharemonthrentprice` типу `DecimalField` з вартістю оренди цього варіанту спільного використання невеликого приміщення на місяць (базова вартість в євро).

Клас `Smallspaceshare` має поле `smallspaceshareyearrentprice` типу `DecimalField` з вартістю оренди цього варіанту спільного використання невеликого приміщення на рік (базова вартість в євро).

Клас `Smallspaceshareaccess` реалізує модель періодів доступності невеликого приміщення для спільного використання.

Клас `Smallspaceshareaccess` має поле `smallspaceshare` типу `ForeignKey`, що реалізує оголошення про спільне використання невеликого приміщення, до якого відноситься цей визначений період доступності невеликого приміщення, зі зв'язком з моделлю `Smallspaceshare`.

Клас `Smallspaceshareaccess` має поле `smallspacesharestartdate` типу `DateTimeField` з початком періоду доступності невеликого приміщення для спільного використання.

Клас `Smallspaceshareaccess` має поле `smallspaceshareenddate` типу `DateTimeField` з завершенням періоду доступності невеликого приміщення для спільного використання.

Клас `Smallspacesharephoto` реалізує модель фотографій варіантів спільного використання за невеликим приміщенням.

Клас `Smallspacesharephoto` має поле `smallspaceshare` типу `ForeignKey`, що реалізує варіант спільного використання за невеликим приміщенням, який показаний на цій фотографії, зі зв'язком з моделлю `Smallspaceshare`.

Клас `Smallspacesharephoto` має поле `smallspacesharephotodata` типу `ImageField` з даними фотографії варіанта спільного використання за невеликим приміщенням.

Клас `Smallspacerequest` реалізує модель запитування спільного використання невеликого приміщення.

Клас `Smallspacerequest` має поле `smallspaceshare` типу `ForeignKey`, що реалізує варіант спільного використання невеликого приміщення, якого стосується це запитування, зі зв'язком з моделлю `Smallspaceshare`.

Клас `Smallspacerequest` має поле `spaceshareuser` типу `ForeignKey`, що реалізує користувача, який запитав спільне використання цього невеликого приміщення, зі зв'язком з моделлю `Spaceshareuser`.

Клас `Smallspacerequest` має поле `smallspacerequeststate` типу `CharField` зі станом запитування спільного використання невеликого приміщення.

Клас `Smallspacerequest` має поле `smallspacerequestdate` типу `DateTimeField` з часом запитування спільного використання невеликого приміщення.

Клас `Smallspacerequest` має поле `smallspacerequestnote` типу `CharField` з нотаткою до запитування спільного використання невеликого приміщення.

Клас `Smallspacerequest` має поле `smallspacesharestartdate` типу `DateTimeField` з часом початку спільного використання невеликого приміщення за цим запитом.

Клас `Smallspacerequest` має поле `smallspaceshareenddate` типу `DateTimeField` з часом закінчення спільного використання невеликого приміщення за цим запитом.

Клас `Smallspacernote` реалізує модель нотування оголошення про спільне використання невеликого приміщення.

Клас `Smallspacernote` має поле типу `ForeignKey`, що реалізує оголошення про спільне використання невеликого приміщення, до якого зроблено це, зі зв'язком з моделлю `Smallspace`.

Клас `Smallspacenote` має поле `spaceshareuser` типу `ForeignKey`, що реалізує користувача, який занотував це оголошення про спільне використання невеликого приміщення, зі зв'язком з моделлю `Spaceshareuser`.

Клас `Smallspacenote` має поле `smallspacenotetext` типу `CharField` з текстом нотатки до оголошення про спільне використання невеликого приміщення.

Клас `Smallspacenote` має поле `smallspacetype` типу `ForeignKey`, що реалізує тип невеликого приміщення для спільного використання, до якого віднесено цю збережену нотатку, зі зв'язком з моделлю `Smallspacetype`.

Клас `Spaceshareuser` реалізує модель користувача.

Клас `Spaceshareuser` має поле `spaceshareraccount` типу `OneToOneField`, що реалізує обліковий запис цього користувача в Django, зі зв'язком з моделлю `User`.

Клас `Spaceshareuser` має поле `spacesharerphone` типу `CharField` з номером телефону користувача.

Клас `Spaceshareuser` має поле `spaceshareradvertisement` типу `CharField` з описом користувача.

Клас `Spaceshareuser` має поле `spacesharerrentreviewmark` типу `DecimalField` з усередненою оцінкою користувача як орендаря спільного використання невеликих приміщень.

Клас `Spaceshareuser` має поле `spacesharerphoto` типу `ImageField` з фотографією користувача.

Клас `Spaceshareuser` має поле `spacecity` типу `ForeignKey`, що реалізує місто, в якому проживає користувач, зі зв'язком з моделлю `Spacecity`.

Клас `Smallspacesharereview` реалізує модель відгуків про досвід спільного використання невеликих приміщень.

Клас `Smallspacesharereview` має поле `smallspacerequest` типу `ForeignKey`, що реалізує запитування спільного використання невеликого приміщення, за яким сформовано цей відгук, зі зв'язком з моделлю `Smallspacerequest`.

Клас `Smallspacesharereview` має поле `smallspacesharereviewdate` типу `DateTimeField` з часом створення відгуку про досвід спільного використання невеликого приміщення.

Клас `Smallspacesharereview` має поле `smallspacesharereviewtext` типу `TextField` з текстом відгуку про досвід спільного використання невеликого приміщення.

Клас `Smallspacesharereview` має поле `smallspacesharereviewscore` типу `PositiveSmallIntegerField` з оцінкою відгуку про досвід спільного використання невеликого приміщення.

Клас `Smallspacesharereview` має поле `smallspacesharereviewside` типу `BooleanField` зі стороною, яка створила відгук про досвід спільного використання невеликого приміщення.

Клас `Spacesharemessage` реалізує модель повідомлення за невеликим приміщенням для спільного використання.

Клас `Spacesharemessage` має поле `smallspace` типу `ForeignKey`, що реалізує оголошення про спільне використання невеликого приміщення, за яким створено це повідомлення, зі зв'язком з моделлю `Smallspace`.

Клас `Spacesharemessage` має поле `spaceshareuser` типу `ForeignKey`, що реалізує користувача, який створив це повідомлення, зі зв'язком з моделлю `Spaceshareuser`.

Клас `Spacesharemessage` має поле `spacesharemessagedate` типу `DateTimeField` з часом створення повідомлення за невеликим приміщенням для спільного використання.

Клас `Spacesharemessage` має поле `spacesharemessagetext` типу `CharField` з текстом повідомлення за невеликим приміщенням для спільного використання.

Функції представлень [9] реалізують саме підтримку виділення даних під реалізацію функціональних можливостей ПЗ спільного використання невеликих приміщень.

Перегляд сторінки власника невеликих приміщень реалізовано функцією `SpaceshareuserOwnerView`, яка має параметр `spaceshareuser_username`, який задає логін користувача як власника невеликих приміщень. Логін може бути не заданий, тоді застосовуються дані авторизації. Основні результати роботи передаються в шаблон вебсторінки `spacesharerowner.html`, заповнюючи його словником з ключами `user` з даними самого власника невеликих приміщень та `smallspaces` з даними його 3 невеликих приміщень. Шаблони вебсторінок створені на основі шаблону [10].

Перегляд оголошень про спільне використання невеликих приміщень власника реалізовано функцією `SpaceshareuserOwnerSpacesView`, яка має параметр `spaceshareuser_username`, який задає логін користувача як власника невеликих приміщень. Основні результати роботи передаються в шаблон вебсторінки `sharerownerspaces.html`, заповнюючи його словником з ключами `user` з даними самого власника невеликих приміщень та `smallspaces` з даними його невеликих приміщень. Невеликі приміщення розподіляються порціями по 10 на сторінці.

Перегляд власних оголошень про спільне використання невеликих приміщень реалізовано функцією `OwnSpaceshareuserSpacesView`. Основні результати роботи передаються в шаблон вебсторінки `ownsharerownerspaces.html`, заповнюючи його словником з ключами `user` з даними самого власника невеликих приміщень та `smallspaces` з даними його невеликих приміщень. Невеликі приміщення розподіляються порціями по 10 на сторінці.

Перегляд оголошення про спільне використання невеликого приміщення реалізовано функцією `SmallSpacesView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення. Основні результати роботи передаються в шаблон вебсторінки `smallspace.html`, заповнюючи його словником з ключем `smallspace` з даними самого невеликого приміщення.

Перегляд відгуків про досвід спільного використання за невеликим приміщенням реалізовано функцією `SmallSpaceReviewsView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення. Основні результати роботи передаються в шаблон вебсторінки `smallspacereviews.html`, заповнюючи його словником з ключем `smallspace` з даними самого невеликого приміщення.

Перегляд відгуків про досвід спільного використання за власником невеликих приміщень реалізовано функцією `SmallSpaceOwnerReviewsView` з параметром `spaceshareuser_username`, який задає логін власника невеликих приміщень. Основні результати роботи передаються в шаблон вебсторінки `ownerssmallspacereviews.html`, заповнюючи його словником з ключем `reviews` з даними відгуків про досвід спільного використання за власником невеликих приміщень.

Перегляд відгуків про досвід взаємодії за спільним використанням за орендарем реалізовано функцією `SmallSpaceRenterReviewsView` з параметром `spaceshareuser_username`, який задає логін орендаря. Основні результати роботи передаються в шаблон вебсторінки `renterssmallspacereviews.html`, заповнюючи його словником з ключем `renter` з даними орендаря та `reviews` з даними відгуків про досвід взаємодії за спільним використанням.

Перегляд сторінки орендаря спільного використання невеликих приміщень реалізовано функцією `SmallSpaceRenterView` з параметром `spaceshareuser_username`, який задає логін орендаря або використовується з даних авторизації, якщо цей логін не задано в значенні параметра. Основні результати роботи передаються в шаблон вебсторінки `renter.html`, заповнюючи його словником з ключами `renter` з даними орендаря спільного використання невеликих приміщень та `reviews` з відгуками про орендаря.

Перегляд невеликих приміщень у використанні користувача реалізовано функцією `SharedSmallSpaceRenterView` з параметром `spaceshareuser_username`. Основні результати роботи передаються в шаблон вебсторінки `rentershare.html`, заповнюючи його словником з ключами `renter` з

даними орендаря спільного використання невеликих приміщень та requests з підтвердженими, але не завершеними запитування орендаря щодо спільного використання невеликих приміщень.

Перегляд запитувань спільного використання за невеликим приміщенням реалізовано функцією `SmallSpaceRequestsView` з параметром `smallspace_id`, який задає ідентифікатор невеликого приміщення. Основні результати роботи передаються в шаблон вебсторінки `smallspacerequests.html`, заповнюючи його словником з ключем `smallspace` з даними запитувань спільного використання за невеликим приміщенням.

Перегляд підтверджених запитувань спільного використання за всіма невеликими приміщеннями власника реалізовано функцією `SmallSpaceApprovedRequestsView`. Основні результати роботи передаються в шаблон вебсторінки `approvedsmallspacerequests.html`, заповнюючи його словником з ключами `smallspaces` з даними невеликих приміщень авторизованого власника та `requests` з даними підтверджених запитувань спільного використання.

Перегляд скасованих запитувань спільного використання за всіма невеликими приміщеннями власника реалізовано функцією `SmallSpaceCancelledRequestsView`. Основні результати роботи передаються в шаблон вебсторінки `cancelledsmallspacerequests.html`, заповнюючи його словником з ключами `smallspaces` з даними невеликих приміщень авторизованого власника та `requests` з даними скасованих запитувань спільного використання.

Перегляд занотованих оголошень про спільне використання невеликих приміщень реалізовано функцією `SmallSpaceNotesView`. Основні результати роботи передаються в шаблон вебсторінки `notes.html`, заповнюючи його словником з ключем `smallspaces` з даними занотованих оголошень про спільне використання невеликих приміщень.

Перегляд всіх запитувань спільного використання невеликих приміщень за орендарем реалізовано функцією

RenterSmallSpaceRequestsView. Дані орендаря беруться з авторизації. Основні результати роботи передаються в шаблон вебсторінки `renterrequests.html`, заповнюючи його словником з ключем `requests` з запитуваннями спільного використання невеликих приміщень.

Створення відгуку про досвід спільного використання невеликого приміщення реалізовано функцією `SmallspaceshareRenterReview` з параметром `smallspacerequest_id`, який задає ідентифікатор запитування спільного використання невеликого приміщення, за яким створюється відгук. Для підтримки функції створено клас форми `SmallspacesharereviewForm`.

Створення відгуку про досвід взаємодії з орендарем спільного використання невеликого приміщення реалізовано функцією `SmallspaceshareOwnerReview` з параметром `smallspacerequest_id`, який задає ідентифікатор запитування спільного використання невеликого приміщення, за яким створюється відгук. Для підтримки функції створено клас форми `SmallspacesharereviewForm`.

Додавання оголошення про спільне використання невеликого приміщення реалізовано функцією `SmallspaceAddView`. Для підтримки функції створено клас форми `SmallspaceForm`. Редагування оголошення про спільне використання невеликого приміщення базується на тій самій формі та функції `SmallspaceEditView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення.

Зняття оголошення про спільне використання невеликого приміщення реалізовано функцією `SmallspaceEndView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення.

Підтвердження запитування спільного використання невеликого приміщення реалізовано функцією `SmallspacerequestApproveView` з параметром `smallspacerequest_id`, який задає ідентифікатор запитування спільного використання невеликого приміщення.

Завершення спільного використання невеликого приміщення реалізовано функцією `SmallspacerequestEndView` з параметром `smallspacerequest_id`, який задає ідентифікатор запитування спільного використання невеликого приміщення.

Перегляд повідомлень за невеликим приміщенням для спільного використання у орендаря реалізовано функцією `SmallSpaceMessagesView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення. Основні результати роботи передаються в шаблон вебсторінки `smallspacemessages.html`, заповнюючи його словником з ключем `messages` з даними повідомлень за невеликим приміщенням для спільного використання у орендаря. Відправлення повідомлення за невеликим приміщенням для спільного використання підтримується всередині цього представлення.

Перегляд невеликих приміщень для спільного використання, за якими здійснювалась переписка, реалізовано функцією `SmallSpaceInMessagesView`. Основні результати роботи передаються в шаблон вебсторінки `smallspacesformessages.html`, заповнюючи його словником з ключем `smallspaces` з даними невеликих приміщень для спільного використання, за якими здійснювалась переписка.

Перегляд переписок за невеликим приміщенням для спільного використання у власника реалізовано функцією `SmallSpaceOwnerMessagesView`. Основні результати роботи передаються в шаблон вебсторінки `smallspacesownermessages.html`, заповнюючи його словником з ключем `smallspaces` з даними невеликих приміщень власника для спільного використання, за якими здійснювалась переписка.

Заповнення користувацьких даних реалізовано функцією `SpaceshareuserSetView` на основі класу форми `SpaceshareuserForm`.

Додавання варіанту спільного використання за невеликим приміщенням реалізовано функцією `SmallspaceshareAddView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання

невеликого приміщення. Для підтримки функції створено клас форми `SmallspaceshareForm`. Редагування варіанту спільного використання за невеликим приміщенням базується на тій самій формі та функції `SmallspaceshareEditView` з параметром `smallspaceshare_id`, який задає ідентифікатор варіанту спільного використання за невеликим приміщенням. Вилучення варіанту спільного використання за невеликим приміщенням реалізовано функцією `SmallspaceshareDeleteView` з параметром `smallspaceshare_id`, який задає ідентифікатор варіанту спільного використання за невеликим приміщенням.

Нотування оголошення про спільне використання невеликого приміщення реалізовано функцією `SmallspacenoteAddView` з параметрами `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення, `smallspacenotetext`, який задає текст нотатки, та `smallspacetype_id`, який задає тип невеликого приміщення для спільного використання.

Відміна нотування оголошення про спільне використання невеликого приміщення реалізовано функцією `SmallspacenoteDeleteView` з параметром `smallspace_id`, який задає ідентифікатор оголошення про спільне використання невеликого приміщення, за яким раніше цим користувачем було створено нотатку.

Додавання періоду доступності невеликого приміщення для спільного використання реалізовано функцією `SmallspaceshareaccessAddView` з параметром `smallspaceshare_id`, який задає ідентифікатор варіанту спільного використання невеликого приміщення, на основі класу форми `SmallspaceshareaccessForm`.

Вилучення періоду доступності невеликого приміщення для спільного використання реалізовано функцією `SmallspaceshareaccessDeleteView` з параметром `smallspaceshareaccess_id`, який задає ідентифікатор періоду доступності невеликого приміщення для спільного використання.

Пошук невеликих приміщень для спільного використання реалізовано функцією `SmallspaceSearchView` з шаблоном вебсторінки `smallspacesearch.html` з визначенням параметрів пошуку через GET-запит.

Запитування спільного використання невеликого приміщення реалізовано функцією `SmallspacerequestAddView` з параметром `smallspaceshare_id`, який задає ідентифікатор варіанту спільного використання невеликого приміщення, на основі класу форми `SmallspacerequestForm`.

Скасування запитування спільного використання невеликого приміщення реалізовано функцією `SmallspacerequestDeleteView` з параметром `smallspacerequest_id`, який задає ідентифікатор запитування спільного використання невеликого приміщення.

3.3 Висновки за розділом 3

У розділі 3 ПЗ спільного використання невеликих приміщень, розроблене за даним етапом, було описане стосовно тих рішень, які були прийняті в процесі реалізації.

4 ЕКСПЛУАТАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМИ

4.1 Призначення програми

ПЗ спільного використання невеликих приміщень може використовуватись як власниками невеликих приміщень, так і потенційними або фактичними орендарями для того, щоб розподілити наявний простір у невеликих приміщеннях для оренди різних користувачів або запропонувати для спільного використання весь наявний простір невеликого приміщення, керувати запитами зацікавлених осіб у такій оренді, шукати варіанти, які влаштовують користувачів.

4.2 Умови виконання програми

ПЗ спільного використання невеликих приміщень задля виконання на сервері потребує мінімально процесора з 2 ядрами, 4 Гб оперативної пам'яті, 100 Гб сховища на жорсткому диску, встановлені інтерпретатор мови Python та систему керування базами даних PostgreSQL.

4.3 Експлуатація програми спільного використання невеликих приміщень

ПЗ спільного використання невеликих приміщень для користувача має загальний інтерфейс, який змінюється тільки з врахуванням того, чи цей користувач авторизувався, чи ні. Якщо він не авторизувався, то він може виконувати пошук варіантів спільного використання невеликих приміщень та перейти безпосередньо до авторизації. Якщо він авторизувався, то йому доступні одразу всі інструменти, які включають як засоби роботи орендаря невеликих приміщень, так і власника.

Будь-яке оголошення про спільне використання невеликих приміщень, яке користувач переглядає, він може занотувати, а тоді відобразити всі

занотовані оголошення через розділ меню «Нотатки» (рис. 4.1). З цього розділу нотатки можна і вилучити. Якщо під час нотування додано текстову примітку, то вона також буде відображатися біля самого оголошення.

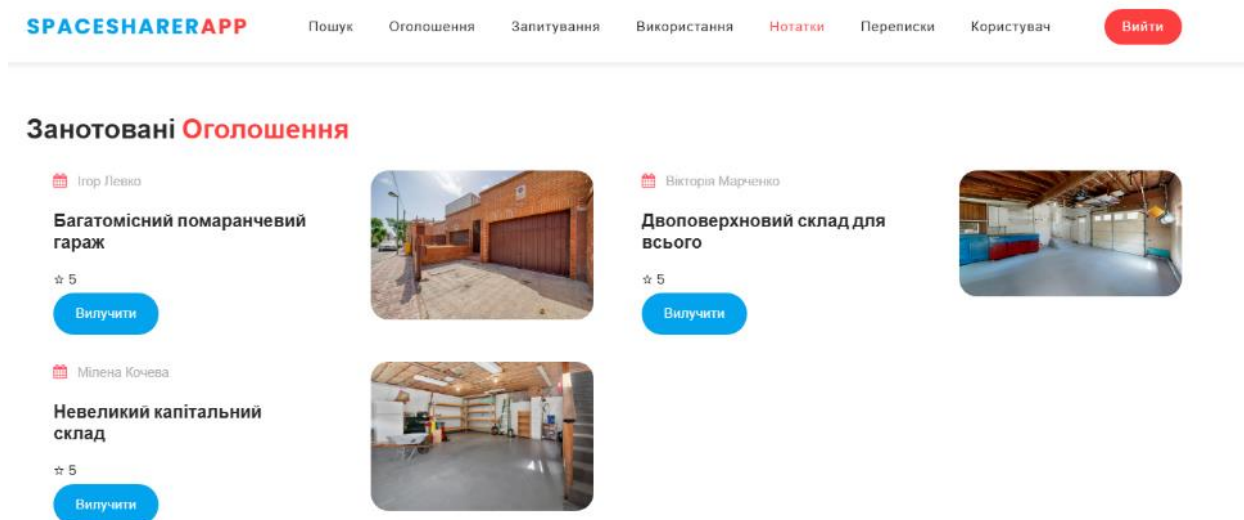


Рисунок 4.1 – Перегляд занотованих оголошень про спільне використання невеликих приміщень

Користувач як власник з розділу меню «Оголошення» має доступ до переліку власних оголошень про спільне використання невеликих приміщень (рис. 4.2). За цим переліком користувач може за кожним оголошенням виконати певні дії за допомогою відповідних кнопок «Додати період доступності», «Вилучити період доступності», які дозволяють відповідно створити новий період доступності для невеликого приміщення або обрати один з існуючих та вилучити його, «Редагувати», щоб перейти до редагування оголошення про спільне використання невеликого приміщення, «Зняти», щоб тимчасово орендарі не мали доступ до роботи з цим невеликим приміщенням. Окрім того доступна можливість додати оголошення за допомогою відповідної кнопки зверху.

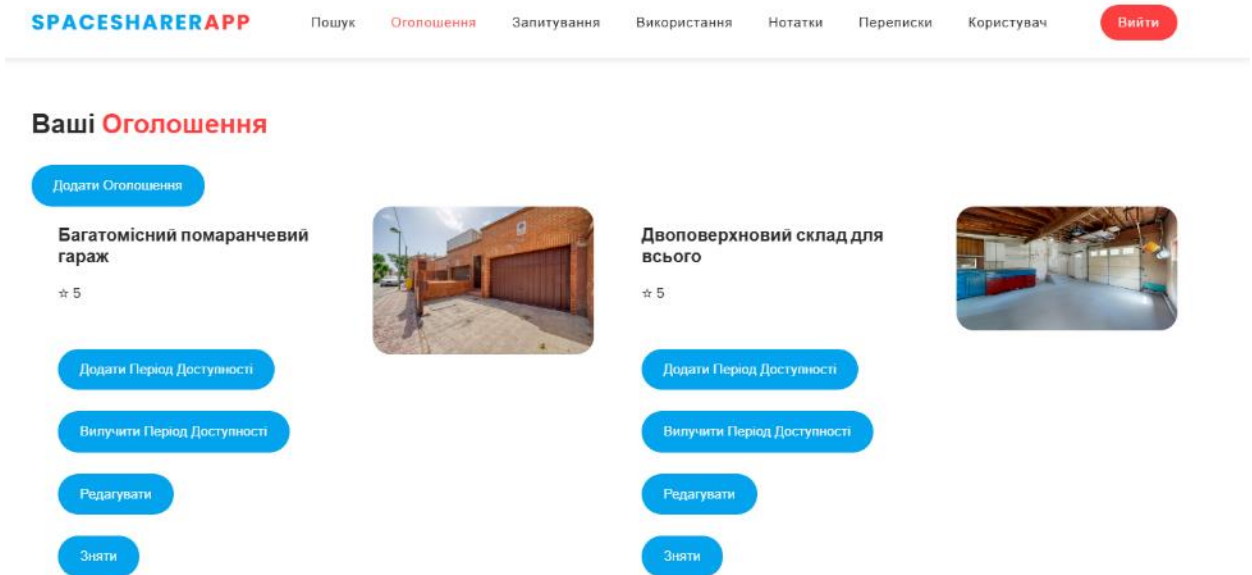


Рисунок 4.2 – Перегляд власних оголошень про спільне використання невеликих приміщень

Під час роботи з програмою за логікою орендаря можна відкрити розділ «Використання», щоб швидко побачити ті оренди, якими вже користується орендар. З розділу «Переписки» доступне переключення за двома режимами, в першому з яких можна переглянути перелік невеликих приміщень, за якими питання власникам відправляв користувач, а в другому перелік власних приміщень, обираючи яке можна перейти до перегляду повідомлень за ним, адресованих користувачу.

Будь-яке оголошення можна повністю відкрити для перегляду деталей (рис. 4.3). Якщо з таким оголошенням працює орендар, то йому доступна основна інформація, перелік варіантів спільного використання невеликого приміщення, з якого він може обрати потрібний для оренди та замовити її, також на сторінці нижче доступний весь набір фотографій, можливість переходу до відправлення повідомлення за відповідним невеликим приміщенням власнику.

У свою чергу власник невеликого приміщення бачить таке оголошення про спільне використання невеликого приміщення подібно, але має можливості керування ним.

SPACESHARERAPP

Пошук

Оголошення

Запитування

Використання

Нотатки

Переписки

Користувач

Вийти



Ігор Левко

Гаражі

☆ 5

Багатомісний помаранчевий гараж

Деповерхновий гараж здатен вмістити доволі велику кількість в тому числі важливих речей, які мають особливі умови зберігання. Ми будемо раді всім орендарям та спробуємо створити комфортні умови для всіх

Обсяг: 750 м3
Ширина: 10 м
Висота: 2.5 м
Довжина: 30 м

В Нотатки

Зона індивідуального збереження

☆ 5

Обсяг: 250 м3

Ширина: 10 м

Висота: 2.5 м

Довжина: 10 м

Тип: приміщення з орендарями

Тип доступу: спільне приміщення

Зона для збереження речей з шафками, холодильником та робочим столом за необхідності

Замовити



Зона довільного збереження

☆ 5

Обсяг: 62.5 м3

Ширина: 5 м

Висота: 2.5 м

Довжина: 5 м

Тип: приміщення з орендарями

Тип доступу: спільне приміщення

Зона для збереження біля самих воріт зі швидким доступом

Замовити



Рисунок 4.3 – Перегляд оголошення про спільне використання невеликого приміщення

Власник невеликого приміщення на сторінці оголошення про спільне використання невеликого приміщення (рис. 4.4) може одразу перейти до його редагування, зняти оголошення з розгляду, додати або вилучити період доступності, додати новий варіант спільного використання цього приміщення. За кожним з раніше доданих варіантів спільного використання невеликого приміщення можна перейти до редагування або вилучення такого варіанта.

Також користувачу доступний розділ «Запитування», звідки відображаються власні запитування, але також є можливість перейти до запитувань за власними невеликим приміщеннями.

SPACESHARERAPP

Пошук

Оголошення

Запитування

Використання

Нотатки

Переписки

Користувач

Вийти

Багатомісний Помаранчевий Гараж

Редагувати

Зняти

Додати Період Доступності

Вилучити Період Доступності

Додати Варіант Використання

Гаражі

☆ 5

Багатомісний помаранчевий гараж

Двоповерховий гараж здатен вмістити доволі велику кількість в тому числі важливих речей, які мають особливі умови зберігання. Ми будемо раді всім орендарям та спробуємо створити комфортні умови для всіх

Обсяг: 750 м3

Ширина: 10 м

Висота: 2.5 м

Довжина: 30 м

Зона індивідуального збереження

☆ 5

Обсяг: 250 м3

Ширина: 10 м

Висота: 2.5 м

Довжина: 10 м

Тип приміщення з орендарями

Тип доступу: спільне приміщення

Зона для збереження речей з шафками, холодильником та робочим столом за необхідності

Редагувати

Вилучити

Зона довільного збереження

☆ 5

Обсяг: 62.5 м3

Ширина: 5 м

Висота: 2.5 м

Довжина: 5 м

Тип приміщення з орендарями

Тип доступу: спільне приміщення

Зона для збереження біля самих воріт зі швидким доступом

Редагувати

Вилучити

Рисунок 4.4 – Перегляд оголошення про спільне використання невеликого приміщення власником

За власними запитуваннями спільного використання невеликого приміщення (рис. 4.5) можна або отримати детальну інформацію, або перейти до скасування такого запитування, або завершення, якщо воно підтверджене, розпочате за часом, але ще не завершилось.

Власник у своєму режимі може переглядати всі запитування, а може робити це за обраним власним невеликим приміщенням.



Рисунок 4.5 – Перегляд всіх запитувань спільного використання невеликих приміщень за орендарем

З розділу «Користувач» можна редагувати власні користувацькі дані, спочатку переглянувши їх.

4.4 Тестування програми спільного використання невеликих приміщень

ПЗ спільного використання невеликих приміщень було протестовано для того, щоб встановити ступінь його відповідності початковим вимогам до програми. Для встановлення ступеня відповідності тестування було проведено за набором функцій, визначених для реалізації програми. За кожною функцією на основі відповідного сценарію використання було прописано очікуваний результат.

Далі під час тестування виконувалась перевірка того, чи відповідає отриманий результат тому результату, який очікувався. За підсумковим тестуванням усі результати за всіма функціями відповідали очікуванім. У такому випадку в табл. 4.1 мали проставлятися позначки «+». У випадку невідповідності, якщо б таке було виявлено, проставлялись мала позначка «-».

Таблиця 4.1 – Результати тестування програми спільного використання невеликих приміщень

Функція	Результат
1	2
Додавання оголошення про спільне використання невеликого приміщення	+
Редагування оголошення про спільне використання невеликого приміщення	+
Зняття оголошення про спільне використання невеликого приміщення	+
Перегляд власних оголошень про спільне використання невеликих приміщень	+
Перегляд невеликих приміщень у використанні користувача	+
Перегляд оголошення про спільне використання невеликого приміщення	+
Нотування оголошення про спільне використання невеликого приміщення	+
Перегляд занотованих оголошень про спільне використання невеликих приміщень	+
Відміна нотування оголошення про спільне використання невеликого приміщення	+
Запитування спільного використання невеликого приміщення	+
Редагування варіанту спільного використання за невеликим приміщенням	+
Вилучення варіанту спільного використання за невеликим приміщенням	+

Продовження таблиці 4.1

1	2
Підтвердження запитування спільного використання невеликого приміщення	+
Завершення спільного використання невеликого приміщення	+
Перегляд сторінки власника невеликих приміщень	+
Перегляд сторінки орендаря спільного використання невеликих приміщень	+
Заповнення користувацьких даних	+
Перегляд оголошень про спільне використання невеликих приміщень власника	+
Створення відгуку про досвід спільного використання невеликого приміщення	+
Перегляд відгуків про досвід спільного використання за невеликим приміщенням	+
Перегляд відгуків про досвід спільного використання за власником невеликих приміщень	+
Створення відгуку про досвід взаємодії з орендарем спільного використання невеликого приміщення	+
Перегляд відгуків про досвід взаємодії за спільним використанням за орендарем	+
Перегляд запитувань спільного використання за невеликим приміщенням	+
Перегляд запитувань спільного використання за всіма невеликими приміщеннями власника	+
Перегляд підтверджених запитувань спільного використання за всіма невеликими приміщеннями власника	+

Кінець таблиці 4.1

1	2
Відправлення повідомлення за невеликим приміщенням для спільного використання	+
Перегляд повідомлень за невеликим приміщенням для спільного використання у орендаря	+
Перегляд невеликих приміщень для спільного використання, за якими здійснювалась переписка	+
Перегляд переписок за невеликим приміщенням для спільного використання у власника	+
Реєстрація облікового запису	+
Авторизація	+
Додавання періоду доступності невеликого приміщення для спільного використання	+
Вилучення періоду доступності невеликого приміщення для спільного використання	+
Скасування запитування спільного використання невеликого приміщення	+
Перегляд скасованих запитувань спільного використання за всіма невеликими приміщеннями власника	+
Перегляд всіх запитувань спільного використання невеликих приміщень за орендарем	+
Пошук невеликих приміщень для спільного використання	+
Додавання варіанту спільного використання за невеликим приміщенням	+

4.5 Висновки за розділом 4

ПЗ спільного використання невеликих приміщень було перевірено на працездатність у даному розділі, в підсумку було створено інструкцію щодо

використання ПЗ користувачами для досягнення своїх завдань. Тестування в підсумку завершилось успішно. ПЗ спільного використання невеликих приміщень відповідає початково встановленим вимогам до нього.

ВИСНОВКИ

У виконаній роботі досягнуто зменшення витрат на оренду невеликих приміщень для зберігання речей шляхом розділення цих витрат між різними користувачами на основі розробки ПЗ спільного використання невеликих приміщень. Для розділення витрат запропоновано кожному власнику визначати сукупність варіантів спільного використання невеликого приміщення, створюючи таким чином окремі комірки, частини, за допомогою яких відділяти простір для різних орендарів у складі одного невеликого приміщення.

Розроблене ПЗ спільного використання невеликих приміщень використовує мову Python, фреймворк Django як основні технології веброзробки, систему PostgreSQL для забезпечення підтримки сховища даних.

Розроблена програма може використовуватись як власниками невеликих приміщень, так і потенційними або фактичними орендарями для того, щоб розподілити наявний простір у невеликих приміщеннях для оренди різних користувачів або запропонувати для спільного використання весь наявний простір невеликого приміщення, керувати запитами зацікавлених осіб у такій оренді, шукати варіанти, які влаштовують користувачів.

Програма дозволяє виконувати додавання, редагування оголошень про спільне використання невеликого приміщення, за необхідності зняття таких оголошень, перегляд власних оголошень про спільне використання невеликих приміщень, перегляд невеликих приміщень у використанні користувача, додавання, редагування, вилучення варіантів спільного використання за невеликим приміщенням, перегляд оголошень про спільне використання невеликого приміщення, їх нотування, перегляд занотованих оголошень, запитування спільного використання невеликого приміщення та скасування, підтвердження таких запитувань, завершення спільного використання невеликого приміщення, перегляд сторінки власника невеликих приміщень, орендаря спільного використання невеликих приміщень, редагування даних

власника невеликих приміщень, орендаря спільного використання невеликих приміщень, створення відгуку про досвід спільного використання невеликого приміщення, перегляд відгуків про досвід спільного використання за невеликим приміщенням, за власником невеликих приміщень, створення відгуку про досвід взаємодії з орендарем спільного використання невеликого приміщення, перегляд відгуків про досвід взаємодії за спільним використанням за орендарем, перегляд запитувачів спільного використання за невеликим приміщенням, за всіма невеликими приміщеннями власника, перегляд підтверджених запитувачів спільного використання за всіма невеликими приміщеннями власника, перегляд незавершених, скасованих запитувачів спільних використань за всіма невеликими приміщеннями власника, перегляд всіх запитувачів спільного використання невеликих приміщень за орендарем, пошук невеликих приміщень для спільного використання, відправлення повідомлення за невеликим приміщенням для спільного використання, перегляд повідомлень за невеликим приміщенням для спільного використання у орендаря, перегляд невеликих приміщень для спільного використання, за якими здійснювалась переписка, перегляд переписок за невеликим приміщенням для спільного використання у власника, додавання, вилучення періоду доступності невеликого приміщення для спільного використання, перегляд оголошень про спільне використання невеликих приміщень власника.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Neighbor. Your Storage & Parking Marketplace. URL: <https://www.neighbor.com/> (date of access: 16.05.2026).
2. Shared Storage. URL: <https://sharedstorage.ca/> (date of access: 16.05.2026).
3. Shared Storage – Apps on Google Play. URL: <https://play.google.com/store/apps/details?id=com.app.sharedstorage> (date of access: 16.05.2026).
4. Download Django. Django. URL: <https://www.djangoproject.com/download/> (date of access: 18.05.2026).
5. Goswami S. Django or Flask? Which Python Web Framework Is Right for Me in 2026? *Cloudways*. URL: <https://www.cloudways.com/blog/django-or-flask/> (date of access: 18.05.2026).
6. GitHub – pallets/flask: The Python micro framework for building web applications. GitHub. URL: <https://github.com/pallets/flask/> (date of access: 18.05.2026).
7. Flask. *PyPI*. <https://pypi.org/project/Flask/> (date of access: 18.05.2026).
8. Models. Django documentation. Django. URL: <https://docs.djangoproject.com/en/6.0/topics/db/models/> (date of access: 20.05.2026).
9. Writing views. Django documentation. Django. URL: <https://docs.djangoproject.com/en/6.0/topics/http/views/> (date of access: 20.05.2026).
10. Free Template 562 Space Dynamic. URL: <https://templatemo.com/tm-562-space-dynamic> (date of access: 20.05.2026).

ДОДАТОК А
Технічне завдання

Вступ

Люди нерідко потребують простору, де вони могли б зберігати речі, таким чином зменшуючи обсяг необхідного простору вдома. Для такого зберігання можуть бути потрібні невеликі приміщення або навіть частини кімнат, гаражів та подібних споруд. Такі приміщення можуть бути навіть розділені на використовуватися різними орендарями в підсумку. Власник може зберігати в частині такого приміщення власні речі, а іншу пропонувати для оренди іншим користувачам. Така логіка є доцільною в умовах, коли оренда такого невеликого приміщення може бути доволі високою і часом дорівнювати вартості оренди повноцінного житла. Щоб знизити цю вартість, спільне використання може бути підходящою моделлю для потенційно великої кількості користувачів. Реалізація такої концепції має актуальність та має бути забезпечена в цій роботі.

A.1 Підстави для розробки

Робота виконана за темою «Програмне забезпечення спільного використання невеликих приміщень» у відповідності з даними Наказу № 139 від 7.04.2026 року в Національному університеті «Запорізька політехніка».

A.2 Призначення розробки

ПЗ спільного використання невеликих приміщень може використовуватись як власниками невеликих приміщень, так і потенційними або фактичними орендарями для того, щоб розподілити наявний простір у невеликих приміщеннях для оренди різних користувачів або запропонувати для спільного використання весь наявний простір невеликого приміщення, керувати запитами зацікавлених осіб у такій оренді, шукати варіанти, які влаштовують користувачів.

A.3 Основні вимоги до програми

A.3.1 Вимоги до функціональних характеристик

ПЗ спільного використання невеликих приміщень має відповідати функціональним характеристикам, серед яких:

- додавання оголошення про спільне використання невеликого приміщення;
- редагування оголошення про спільне використання невеликого приміщення;
- зняття оголошення про спільне використання невеликого приміщення;
- перегляд власних оголошень про спільне використання невеликих приміщень;
- перегляд невеликих приміщень у використанні користувача;
- додавання варіанту спільного використання за невеликим приміщенням;
- редагування варіанту спільного використання за невеликим приміщенням;
- вилучення варіанту спільного використання за невеликим приміщенням;
- перегляд оголошення про спільне використання невеликого приміщення;
- нотування оголошення про спільне використання невеликого приміщення;
- перегляд занотованих оголошень про спільне використання невеликих приміщень;
- відміна нотування оголошення про спільне використання невеликого приміщення;
- запитування спільного використання невеликого приміщення;
- скасування запитування спільного використання невеликого

приміщення;

- підтвердження запитування спільного використання невеликого

приміщення;

- завершення спільного використання невеликого приміщення;

- перегляд сторінки власника невеликих приміщень;

– перегляд сторінки орендаря спільного використання невеликих приміщень;

- редагування даних власника невеликих приміщень;

– редагування даних орендаря спільного використання невеликих приміщень;

- створення відгуку про досвід спільного використання невеликого

приміщення;

- перегляд відгуків про досвід спільного використання за невеликим

приміщенням;

- перегляд відгуків про досвід спільного використання за власником

невеликих приміщень;

- створення відгуку про досвід взаємодії з орендарем спільного

використання невеликого приміщення;

- перегляд відгуків про досвід взаємодії за спільним використанням за

орендарем;

- перегляд запитувань спільного використання за невеликим

приміщенням;

- перегляд запитувань спільного використання за всіма невеликими

приміщеннями власника;

- перегляд підтверджених запитувань спільного використання за всіма

невеликими приміщеннями власника;

- перегляд незавершених спільних використань за всіма невеликими

приміщеннями власника;

- перегляд скасованих запитувань спільного використання за всіма

невеликими приміщеннями власника;

- перегляд всіх запитувань спільного використання невеликих приміщень за орендарем;
- пошук невеликих приміщень для спільного використання;
- відправлення повідомлення за невеликим приміщенням для спільного використання;
- перегляд повідомлень за невеликим приміщенням для спільного використання у орендаря;
- перегляд невеликих приміщень для спільного використання, за якими здійснювалась переписка;
- перегляд переписок за невеликим приміщенням для спільного використання у власника;
- реєстрація облікового запису;
- авторизація;
- додавання періоду доступності невеликого приміщення для спільного використання;
- вилучення періоду доступності невеликого приміщення для спільного використання;
- перегляд оголошень про спільне використання невеликих приміщень власника.

A.3.2 Умови експлуатації

Експлуатаційні вимоги до програми спільного використання невеликих приміщень включають виділення серверу для роботи з програмою шляхом замовлення хостингу. Сервер повинен забезпечувати роботу з системою керування базами даних, мовою програмування, що застосовуватимуться для роботи з програмою після прийняття цих рішень в даній роботі.

A.3.3 Вимоги до складу та параметрів технічних засобів

Для забезпечення хостингу шляхом оренди сервера необхідно враховувати, що мінімальними вимогами, які на сервері виділені виключно під роботу з застосунком підтримки проведення майстер-класів з малювання передбачають наявність процесора мінімально з 2 ядрами, 4 Гб оперативної пам'яті, 100 Гб сховища на жорсткому диску.

A.4 Порядок контролю та приймання

Робота над дипломною кваліфікаційною роботою має виконуватися протягом 8-тижневого циклу, під час якого має бути виконано саму роботу, проведено перевірку роботи на нормоконтроль, проведено процедури рецензування та захисту роботи.

ДОДАТОК Б
Текст програми

```

from django.shortcuts import render
from .models import Spacecountry,
    Spacecity,
    Smallspacetype,
    Smallspace,
    Smallspacephoto,
    Smallspaceshare,
    Smallspaceshareaccess,
    Smallspacesharephoto,
    Smallspacerequest,
    Smallspacenote,
    Spaceshareuser,
    Smallspacesharereview,
    Spacesharemessage
from django.core.paginator import Paginator

#Перегляд сторінки власника невеликих приміщень
def SpaceshareuserOwnerView (request, spaceshareuser_username =
None):
    if spaceshareuser_username:
        spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount__username =
spaceshareuser_username)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Такого власника немає"
                }
            )
        else:
            spaceshare_user = spaceshare_user[0]
    else:
        spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount = request.user)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Треба авторизуватись"
                }
            )

```

```

        }
    )
    else:
        spaceshare_user = spaceshare_user[0]
        small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(smallspaceowner =
spaceshare_user)[:3]
        auth = False
        if spaceshareuser_username:
            spaceshare_user_auth = spaceshare_user
        else:
            spaceshare_user_auth =
Spaceshareuser.objects.filter(spaceshareraccount = request.user)
            if spaceshare_user_auth:
                spaceshare_user_auth = spaceshare_user_auth[0]
                auth = True
            for small_space in small_spaces:
                small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)
                small_space.shares =
Smallspaceshare.objects.filter(smallspace = small_space).count()
                small_space.noted = None
                if auth:
                    smallspace_note =
Smallspacenote.objects.filter(smallspace = small_space, spaceshareuser
= spaceshare_user_auth)
                    if smallspace_note:
                        small_space.noted = True
            return render(
                request,
                "spacesharerapp/spacesharerowner.html",
                {
                    "user": spaceshare_user,
                    "smallspaces": small_spaces
                }
            )

```

#Перегляд оголошень про спільне використання невеликих приміщень
власника

```

def SpaceshareuserOwnerSpacesView (request,
spaceshareuser_username):

```

```

        spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount__username =
spaceshareuser_username)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Такого власника немає"
                }
            )
        else:
            spaceshare_user = spaceshare_user[0]
            small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(smallspaceowner =
spaceshare_user)
            small_spaces = Paginator(small_spaces, 10)
            small_spaces_for_page = request.GET.get('page')
            small_spaces = small_spaces.get_page(small_spaces_for_page)
            auth = False
            spaceshare_user_auth =
Spaceshareuser.objects.filter(spaceshareraccount = request.user)
            if spaceshare_user_auth:
                spaceshare_user_auth = spaceshare_user_auth [0]
                auth = True
            for small_space in small_spaces:
                small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)
                small_space.shares =
Smallspaceshare.objects.filter(smallspace = small_space).count()
                small_space.noted = None
                if auth:
                    smallspace_note =
Smallspacenote.objects.filter(smallspace = small_space, spaceshareuser
= spaceshare_user_auth)
                    if smallspace_note:
                        small_space.noted = True
            return render(
                request,
                "spacesharerapp/sharerownerspaces.html",
                {
                    "user": spaceshare_user,

```

```

        "smallspaces": small_spaces
    }
)

#Перегляд власних оголошень про спільне використання невеликих
приміщень
def OwnSpaceshareuserSpacesView (request):
    spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Треба авторизуватись"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(smallspaceowner =
spaceshare_user)
        small_spaces = Paginator(small_spaces, 10)
        small_spaces_for_page = request.GET.get('page')
        small_spaces = small_spaces.get_page(small_spaces_for_page)
        for small_space in small_spaces:
            small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)
            small_space.shares =
Smallspaceshare.objects.filter(smallspace = small_space).count()
            small_space.noted = None
            smallspace_note =
Smallspacenote.objects.filter(smallspace = small_space, spaceshareuser
= spaceshare_user_auth)
            if smallspace_note:
                small_space.noted = True
        return render(
            request,
            "spacesharerapp/ownsharerownerspaces.html",
            {
                "user": spaceshare_user,
                "smallspaces": small_spaces
            }
        )

```

```

    }
)

#Перегляд оголошення про спільне використання невеликого
приміщення
def SmallSpacesView (request, smallspace_id):
    spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Треба авторизуватись"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        small_space =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(id = smallspace_id)
        small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)
        small_space_shares =
Smallspaceshare.objects.filter(smallspace = small_space)
        import datetime
        share_time = datetime.datetime.now()
        for small_space_share in small_space_shares:
            small_space_share.photos =
Smallspacesharephoto.objects.filter(smallspaceshare =
small_space_share)
            small_space_share.access =
Smallspaceshareaccess.objects.filter(smallspaceshare =
small_space_share,
                smallspacesharestartdate__gte = share_time)
            small_space_share.requests_finished =
Smallspacerequest.objects.filter(smallspaceshare = small_space_share,
                smallspaceshareenddate__gte = share_time,
                smallspacerequeststate = "F")
            small_space_share.requests_approved =
Smallspacerequest.objects.filter(smallspaceshare = small_space_share,
                smallspaceshareenddate__gte = share_time,
                smallspacerequeststate = "A")

```

```

        small_space.shares = small_space_shares
        small_space.noted = None
        smallspace_note = Smallspacenote.objects.filter(smallspace =
small_space, spaceshareuser = spaceshare_user_auth)
        if smallspace_note:
            small_space.noted = True
        small_space_reviews =
Smallspacesharereview.objects.select_related("smallspacerequest").sele
ct_related("smallspacerequest__smallspaceshare").select_related("small
spacerequest__spaceshareuser").filter(smallspacerequest__smallspacesha
re = small_space).order_by("-smallspacesharereviewdate")
        small_space.reviews_count = small_space_reviews.count()
        small_space.reviews = small_space_reviews[:3]
        return render(
            request,
            "spacesharerapp/smallspace.html",
            {
                "smallspace": small_space
            }
        )

```

#Перегляд відгуків про досвід спільного використання за
невеликим приміщенням

```

def SmallSpaceReviewsView (request, smallspace_id):
    small_space =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(id = smallspace_id)
    small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)[0]
    small_space_reviews =
Smallspacesharereview.objects.select_related("smallspacerequest").sele
ct_related("smallspacerequest__smallspaceshare").select_related("small
spacerequest__spaceshareuser").filter(smallspacerequest__smallspacesha
re = small_space).order_by("-smallspacesharereviewdate")
    small_space.reviews_count = small_space_reviews.count()
    small_space_reviews = Paginator(small_space_reviews, 10)
    small_space_reviews_for_page = request.GET.get('page')
    small_space_reviews =
small_space_reviews.get_page(small_space_reviews_for_page)
    small_space.reviews = small_space_reviews
    return render(
        request,
        "spacesharerapp/smallspacereviews.html",

```

```

        {
            "smallspace": small_space
        }
    )

#Перегляд відгуків про досвід спільного використання за
власником невеликих приміщень
def SmallSpaceOwnerReviewsView (request,
spaceshareuser_username):
    spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount__username =
spaceshareuser_username)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Такого власника немає"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(smallspaceowner =
spaceshare_user)
        small_space_reviews =
Smallspacesharereview.objects.select_related("smallspacerequest").sele
ct_related("smallspacerequest__smallspaceshare").select_related("small
spacerequest__spaceshareuser").filter(smallspacerequest__smallspacesha
re__in = small_spaces).order_by("-smallspacesharereviewdate")
        small_space_reviews_count = small_space_reviews.count()
        small_space_reviews = Paginator(small_space_reviews, 10)
        small_space_reviews_for_page = request.GET.get('page')
        small_space_reviews =
small_space_reviews.get_page(small_space_reviews_for_page)
        for small_space_review in small_space_reviews:
            small_space_review.photo =
Smallspacephoto.objects.filter(smallspace =
small_space_review.smallspacerequest.smallspaceshare)[0]
        return render(
            request,
            "spacesharerapp/ownerssmallspacereviews.html",

```

```

        {
            "reviews": small_space_reviews
        }
    )

#Перегляд відгуків про досвід взаємодії за спільним
використанням за орендарем
def SmallSpaceRenterReviewsView (request,
spaceshareuser_username = None):
    if spaceshareuser_username:
        spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount__username =
spaceshareuser_username)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Такого власника немає"
                }
            )
        else:
            spaceshare_user = spaceshare_user[0]
    else:
        spaceshare_user =
Spaceshareuser.objects.filter(spaceshareraccount = request.user)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Авторизуйтеся спочатку"
                }
            )
        else:
            spaceshare_user = spaceshare_user[0]
            small_space_reviews =
Smallspacesharereview.objects.select_related("smallspacerequest").sele
ct_related("smallspacerequest__smallspaceshare").select_related("small
spacerequest__spaceshareuser").filter(smallspacerequest__spaceshareuse
r = spaceshare_user).order_by("-smallspacesharereviewdate")
            small_space_reviews_count = small_space_reviews.count()
            small_space_reviews = Paginator(small_space_reviews, 10)

```

```

        small_space_reviews_for_page = request.GET.get('page')
        small_space_reviews =
small_space_reviews.get_page(small_space_reviews_for_page)
        for small_space_review in small_space_reviews:
            small_space_review.photo =
Smallspacephoto.objects.filter(smallspace =
small_space_review.smallspacerequest.smallspaceshare)[0]
        return render(
            request,
            "spacesharerapp/renterssmallspacereviews.html",
            {
                "renter": spaceshare_user,
                "reviews": small_space_reviews
            }
        )

```

#Перегляд сторінки орендаря спільного використання невеликих приміщень

```

def SmallSpaceRenterView (request, spaceshareuser_username =
None):
    if spaceshareuser_username:
        spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount__username = spaceshareuser_username)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Такого власника немає"
                }
            )
        else:
            spaceshare_user = spaceshare_user[0]
    else:
        spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {

```

```

        "error": "Авторизуйтеся спочатку"
    }
    )
    else:
        spaceshare_user = spaceshare_user[0]
        small_space_reviews =
Smallspacesharereview.objects.select_related("smallspacerequest").select_
ct_related("smallspacerequest__smallspaceshare").select_related("small
spacerequest__spaceshareuser").filter(smallspacerequest__spaceshareuse
r = spaceshare_user).order_by("-smallspacesharereviewdate")
        small_space_reviews_count = small_space_reviews.count()
        small_space_reviews = Paginator(small_space_reviews, 10)
        small_space_reviews_for_page = request.GET.get('page')
        small_space_reviews =
small_space_reviews.get_page(small_space_reviews_for_page)
        for small_space_review in small_space_reviews:
            small_space_review.photo =
Smallspacephoto.objects.filter(smallspace =
small_space_review.smallspacerequest.smallspaceshare)[0]
        return render(
            request,
            "spacesharerapp/renter.html",
            {
                "renter": spaceshare_user,
                "reviews": small_space_reviews
            }
        )

#Перегляд невеликих приміщень у використанні користувача
def SharedSmallSpaceRenterView (request):
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]

```

```

import datetime
share_time = datetime.datetime.now()
small_space_requests_approved =
Smallspacerequest.objects.select_related("smallspaceshare").select_rela
ted("smallspaceshare__smallspace").select_related("smallspaceshare__s
mallspace__smallspacetype").select_related("smallspaceshare__smallspac
e__spacecity").select_related("smallspaceshare__smallspace__smallspace
owner").filter(spaceshareuser = spaceshare_user,
                smallspaceshareenddate__gte = share_time,
                smallspacerequeststate = "A")
for small_space in small_space_requests_approved:
    small_space_review.photo =
Smallspacephoto.objects.filter(smallspace = small_space)[0]
return render(
    request,
    "spacesharerapp/rentershares.html",
    {
        "renter": spaceshare_user,
        "requests":
small_space_requests_approved
    }
)

```

Перегляд запитувань спільного використання за невеликим приміщенням

```

def SmallSpaceRequestsView (request, smallspace_id):
    small_space =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(id = smallspace_id)
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        if small_space.smallspaceowner != spaceshare_user:

```

```

        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Немає прав доступу"
            }
        )
    import datetime
    share_time = datetime.datetime.now()
    small_space_requests =
    Smallspacerequest.objects.select_related("smallspaceshare").select_rela
    ted("smallspaceshare__smallspace").select_related("smallspaceshare__s
    mallspace__smallspacetype").select_related("smallspaceshare__smallspac
    e__spacecity").select_related("smallspaceshare__smallspace__smallspace
    owner").filter(spaceshareuser = spaceshare_user,
                    smallspaceshareenddate__gte =
    share_time).order_by("-smallspacerequestdate")
    small_space.photos =
    Smallspacephoto.objects.filter(smallspace = small_space)[0]
    small_space.requests_count = small_space_requests.count()
    small_space_requests = Paginator(small_space_requests, 10)
    small_space_requests_for_page = request.GET.get('page')
    small_space_requests =
    small_space_requests.get_page(small_space_requests_for_page)
    small_space.requests = small_space_requests
    return render(
        request,
        "spacesharerapp/smallspacerequests.html",
        {
            "smallspace": small_space
        }
    )

# Перегляд запитувань спільного використання за всіма невеликими
приміщеннями власника
def SmallSpaceRequestsView (request):
    spaceshare_user =
    Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
    ccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",

```

```

        {
            "error": "Авторизуйтеся спочатку"
        }
    )
else:
    spaceshare_user = spaceshare_user[0]
    small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("spacecity").select_related("smallspaceowner").filter(smallspaceowner = spaceshare_user)
    import datetime
    share_time = datetime.datetime.now()
    small_space_requests =
Smallspacerequest.objects.select_related("smallspaceshare").select_related("smallspaceshare__smallspace").select_related("smallspaceshare__smallspace__smallspacetype").select_related("smallspaceshare__smallspace__spacecity").select_related("smallspaceshare__smallspace__smallspaceowner").filter(smallspaceshare__smallspace__in = small_spaces,
        smallspaceshareenddate__gte =
share_time).order_by("-smallspacerequestdate")
    for small_space in small_spaces:
        small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)[0]
        small_space.requests_count = small_space_requests.count()
        small_space_requests = Paginator(small_space_requests, 10)
        small_space_requests_for_page = request.GET.get('page')
        small_space_requests =
small_space_requests.get_page(small_space_requests_for_page)
        small_space.requests = small_space_requests
        return render(
            request,
            "spacesharerapp/smallspacerequests.html",
            {
                "smallspaces": small_spaces,
                "requests": small_space_requests
            }
        )

# Перегляд підтверджених запитувань спільного використання за
всіма невеликими приміщеннями власника
def SmallSpaceApprovedRequestsView (request):

```

```

        spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
        if not spaceshare_user:
            return render(
                request,
                "spacesharerapp/error.html",
                {
                    "error": "Авторизуйтесь спочатку"
                }
            )
        else:
            spaceshare_user = spaceshare_user[0]
            small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(smallspaceowner =
spaceshare_user)
            import datetime
            share_time = datetime.datetime.now()
            small_space_requests =
Smallspacerequest.objects.select_related("smallspaceshare").select_rel
ated("smallspaceshare__smallspace").select_related("smallspaceshare__s
mallspace__smallspacetype").select_related("smallspaceshare__smallspac
e__spacecity").select_related("smallspaceshare__smallspace__smallspace
owner").filter(smallspaceshare__smallspace__in = small_spaces,
                smallspaceshareenddate__gte = share_time,
                smallspacerequeststate = "A").order_by("-
smallspacerequestdate")
            for small_space in small_spaces:
                small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)[0]
                small_space.requests_count = small_space_requests.count()
                small_space_requests = Paginator(small_space_requests, 10)
                small_space_requests_for_page = request.GET.get('page')
                small_space_requests =
small_space_requests.get_page(small_space_requests_for_page)
                small_space.requests = small_space_requests
                return render(
                    request,

                    "spacesharerapp/approvedsmallspacerequests.html",
                    {
                        "smallspaces": small_spaces,

```

```

        "requests": small_space_requests
    }
)

#Перегляд скасованих запитувань спільного використання за всіма
невеликими приміщеннями власника
def SmallSpaceCancelledRequestsView (request):
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        small_spaces =
Smallspace.objects.select_related("smallspacetype").select_related("sp
acecity").select_related("smallspaceowner").filter(smallspaceowner =
spaceshare_user)
        import datetime
        share_time = datetime.datetime.now()
        small_space_requests =
Smallspacerequest.objects.select_related("smallspaceshare").select_rel
ated("smallspaceshare__smallspace").select_related("smallspaceshare__s
mallspace__smallspacetype").select_related("smallspaceshare__smallspac
e__spacecity").select_related("smallspaceshare__smallspace__smallspace
owner").filter(smallspaceshare__smallspace__in = small_spaces,
                smallspaceshareenddate__gte = share_time,
                smallspacerequeststate = "C").order_by("-
smallspacerequestdate")
        for small_space in small_spaces:
            small_space.photos =
Smallspacephoto.objects.filter(smallspace = small_space)[0]
            small_space.requests_count = small_space_requests.count()
            small_space_requests = Paginator(small_space_requests, 10)
            small_space_requests_for_page = request.GET.get('page')
            small_space_requests =
small_space_requests.get_page(small_space_requests_for_page)

```

```

        small_space.requests = small_space_requests
        return render(
            request,

            "spacesharerapp/cancelledsmallspacerequests.html",
            {
                "smallspaces": small_spaces,
                "requests": small_space_requests
            }
        )

#Перегляд занотованих оголошень про спільне використання
невеликих приміщень
def SmallSpaceNotesView (request):
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        smallspace_types = Smallspacetype.objects.all()
        for smallspace_type in smallspace_types:
            small_space_notes =
Smallspacenote.objects.select_related("smallspace").select_related("sp
aceshareuser").select_related("smallspace__smallspacetype").select_rel
ated("smallspace__spacecity").select_related("smallspace__spacecity__s
pacecountry").select_related("smallspace__smallspaceowner").filter(spa
ceshareuser = spaceshare_user, smallspacetype = smallspace_type)
            for small_space_note in small_space_notes:
                small_space_note.photo =
Smallspacephoto.objects.filter(smallspace =
small_space_note.smallspace)[0]
                smallspace_type.notes = small_space_notes
        return render(
            request,
            "spacesharerapp/notes.html",

```

```

        {
            "smallspaces": smallspace_types
        }
    )

#Перегляд всіх запитувань спільного використання невеликих
приміщень за орендарем
def RenterSmallSpaceRequestsView (request):
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    if not spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    else:
        spaceshare_user = spaceshare_user[0]
        smallspace_requests =
Smallspacerequest.objects.select_related("smallspaceshare").select_rel
ated("smallspaceshare__smallspace").select_related("smallspaceshare__s
mallspace__smallspacetype").select_related("smallspaceshare__smallspac
e__spacecity").select_related("smallspaceshare__smallspace__smallspace
owner").filter(spaceshareuser = spaceshare_user).order_by("-
smallspacerequestdate")
        small_space_requests = Paginator(smallspace_requests, 10)
        small_space_requests_for_page = request.GET.get('page')
        smallspace_requests =
small_space_requests.get_page(small_space_requests_for_page)
        for smallspace_request in smallspace_requests:
            smallspace_request.photo =
Smallspacephoto.objects.filter(smallspace =
smallspace_request.smallspaceshare.smallspace)[0]
        return render(
            request,
            "spacesharerapp/renterrequests.html",
            {
                "requests": smallspace_requests
            }
        )

```

```

from .forms import SmallspacesharereviewForm

# Створення відгуку про досвід спільного використання невеликого
приміщення
def SmallspaceshareRenterReview (request, smallspacerequest_id):
    smallspace_request =
Smallspacerequest.objects.select_related("smallspaceshare").select_rel
ated("smallspaceshare__smallspace").select_related("smallspaceshare__s
mallspace__smallspacetype").select_related("smallspaceshare__smallspac
e__spacecity").select_related("smallspaceshare__smallspace__smallspace
owner").select_related("spaceshareuser").filter(id =
smallspacerequest_id)
    if not request.user.is_authenticated:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    spaceshare_user = spaceshare_user[0]
    import datetime
    share_time = datetime.datetime.now()
    if smallspace_request.spaceshareuser != spaceshare_user:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Недостатньо прав"
            }
        )
    if request.method == "POST":
        smallspace_share_review_form =
SmallspacesharereviewForm(request.POST)
        if smallspace_share_review_form.is_valid():
            smallspace_share_review =
smallspace_share_review_form.save(commit=False)
            smallspace_share_review.smallspacerequest =
smallspace_request

```

```

        smallspace_share_review.smallspacesharereviewside =
False
        smallspace_share_review.smallspacesharereviewdate =
share_time
        smallspace_share_review.save()
        return redirect(request.path_info)
    else:
        smallspace_share_review_form =
SmallspacesharereviewForm()
        return render(request,
"spacesharerapp/smallspacesharerenterreview.html",
        {
            "form": smallspace_share_review_form
        })

#Створення відгуку про досвід взаємодії з орендарем спільного
використання невеликого приміщення
def SmallspaceshareOwnerReview (request, smallspacerequest_id):
    smallspace_request =
Smallspacerequest.objects.select_related("smallspaceshare").select_rel
ated("smallspaceshare__smallspace").select_related("smallspaceshare__s
mallspace__smallspacetype").select_related("smallspaceshare__smallspac
e__spacecity").select_related("smallspaceshare__smallspace__smallspace
owner").select_related("spaceshareuser").filter(id =
smallspacerequest_id)
    if not request.user.is_authenticated:
        return render(
            request,
            "spacesharerapp/error.html",
            {
                "error": "Авторизуйтеся спочатку"
            }
        )
    spaceshare_user =
Spaceshareuser.objects.select_related("spacecity").filter(spacesharera
ccount = request.user)
    spaceshare_user = spaceshare_user[0]
    import datetime
    share_time = datetime.datetime.now()
    if
smallspace_request.smallspaceshare.smallspace.smallspaceowner !=
spaceshare_user:
        return render(

```

```

        request,
        "spacesharerapp/error.html",
        {
            "error": "Недостатньо прав"
        }
    )
    if request.method == "POST":
        smallspace_share_review_form =
SmallspacesharereviewForm(request.POST)
        if smallspace_share_review_form.is_valid():
            smallspace_share_review =
smallspace_share_review_form.save(commit=False)
            smallspace_share_review.smallspacerequest =
smallspace_request
            smallspace_share_review.smallspacesharereviewside =
True
            smallspace_share_review.smallspacesharereviewdate =
share_time
            smallspace_share_review.save()
            return redirect(request.path_info)
    else:
        smallspace_share_review_form =
SmallspacesharereviewForm()
        return render(request,
"spacesharerapp/smallspaceshareownerreview.html",
        {
            "form": smallspace_share_review_form
        })

```

ДОДАТОК В
Слайди презентації

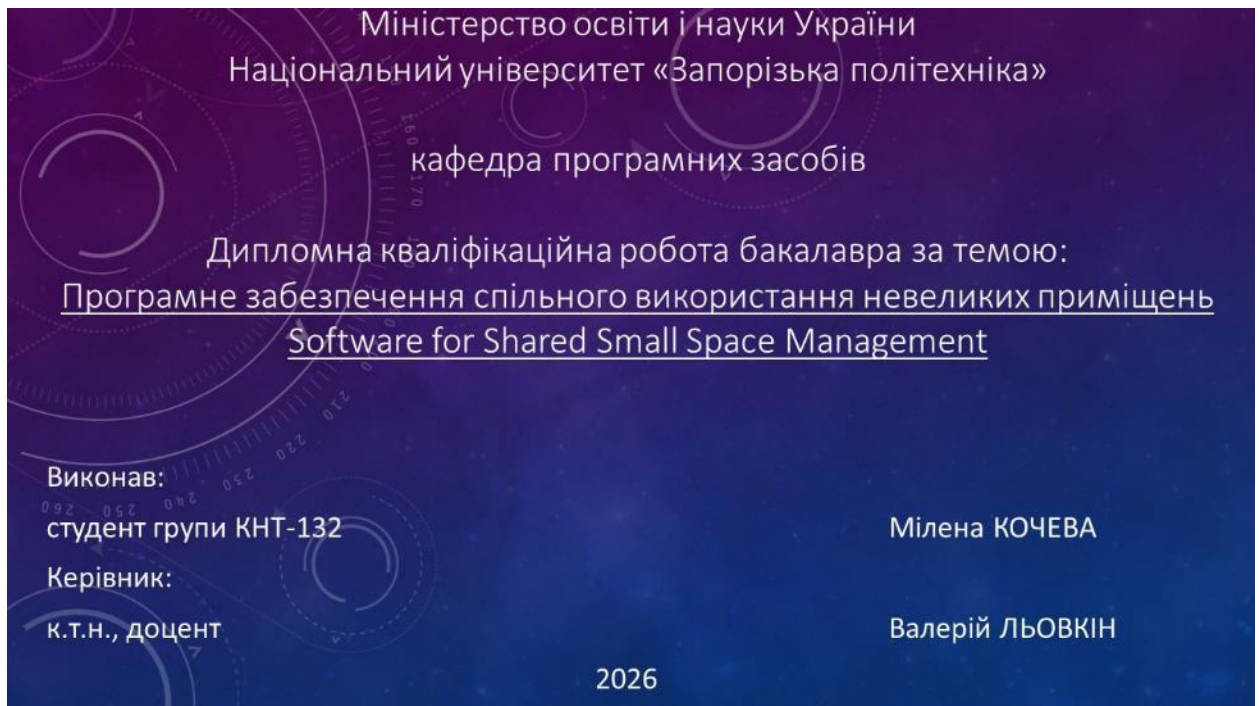


Рисунок В.1 – Початковий слайд

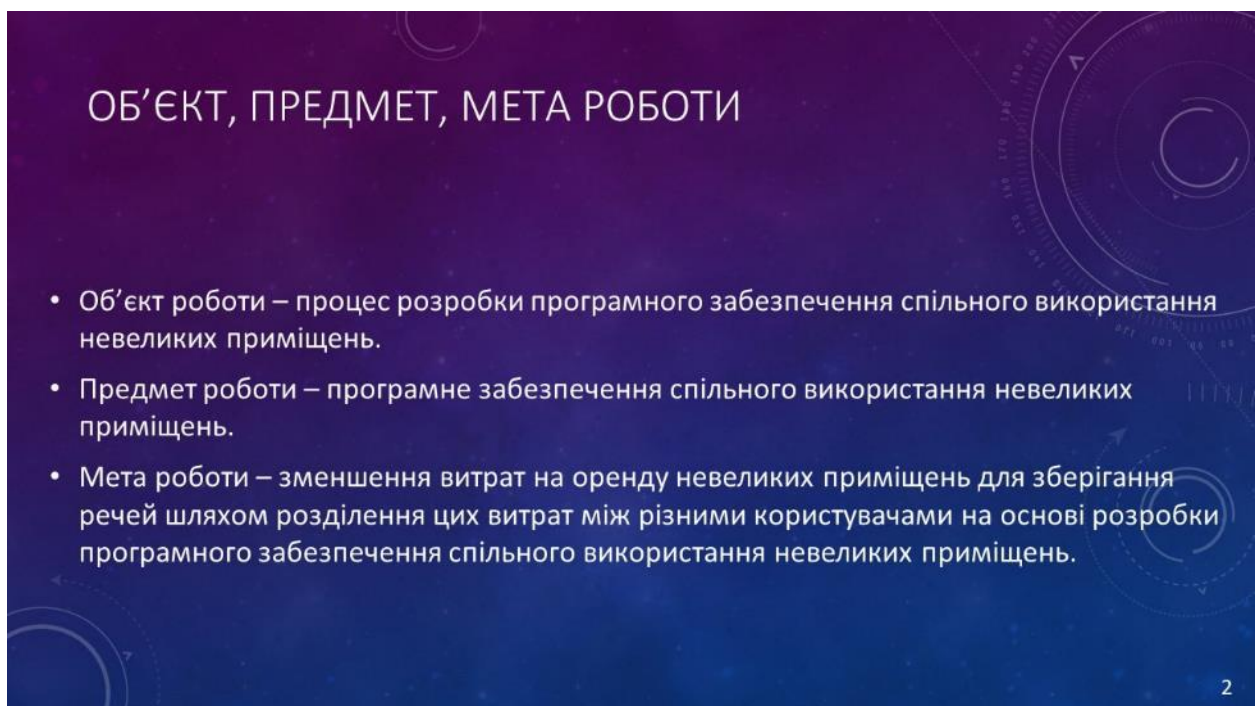


Рисунок В.2 – Об'єкт, предмет, мета роботи

ЗАВДАННЯ РОБОТИ

- аналіз проблеми розробки програмного забезпечення спільного використання невеликих приміщень;
- проектування програмного забезпечення спільного використання невеликих приміщень;
- розробка програмного забезпечення спільного використання невеликих приміщень.

3

Рисунок В.3 – Завдання роботи

ПОРІВНЯННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ

Критерій	Neighbor	Shared Storage	Розробка
Оренда невеликих приміщень	Так	Так	Так
Спільне використання невеликих приміщень	Однакова модель для всіх складових	Ні	Індивідуальна модель для всіх складових
Базова одиниця періоду оренди невеликих приміщень	Місяць	Місяць	День, місяць, рік
Створення відгуків за варіантом використання невеликого приміщення	Ні	Ні	Так

4

Рисунок В.4 – Порівняння програмного забезпечення спільного використання невеликих приміщень

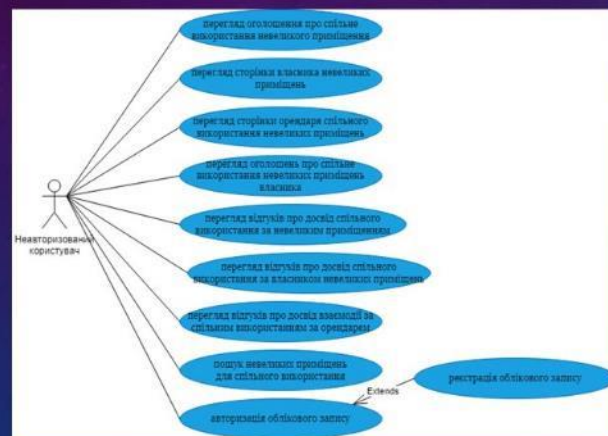
ПОРІВНЯННЯ ФРЕЙМВОРКІВ ДЛЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ

Критерій	Django	Flask
Час розробки	Менший за рахунок вбудованих засобів	Більший
Сценарії використання	Великі застосунки, соціальні платформи	Малі застосунки, створення прототипів
Система аутентифікації	Вбудована	Має бути вибрана і встановлена
Робота з даними	Вбудована система адміністрування, об'єктно-реляційне відображення для бази даних	Засоби мають бути вибрані і встановлені

5

Рисунок В.5 – Порівняння фреймворків для розробки програмного забезпечення спільного використання невеликих приміщень

ДІАГРАМА ВАРІАНТІВ ВИКОРИСТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ НЕАВТОРИЗОВАНИМ КОРИСТУВАЧЕМ



6

Рисунок В.6 – Діаграма варіантів використання програмного забезпечення спільного використання невеликих приміщень неавторизованим користувачем

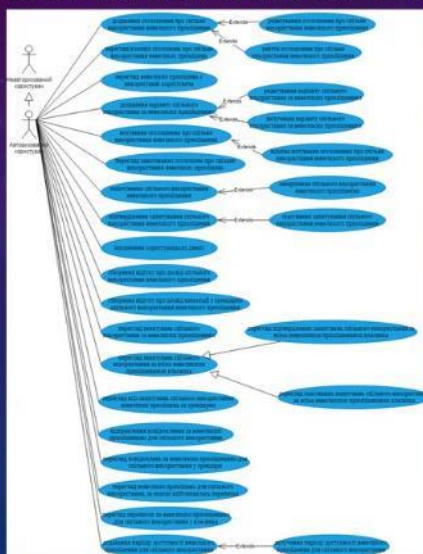


СХЕМА БАЗИ ДАНИХ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ

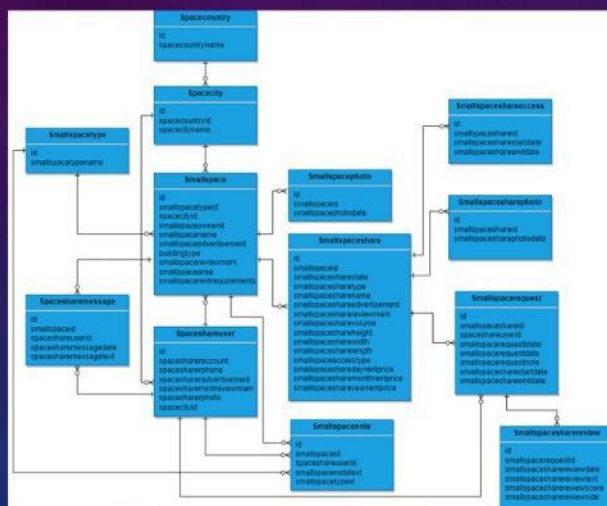
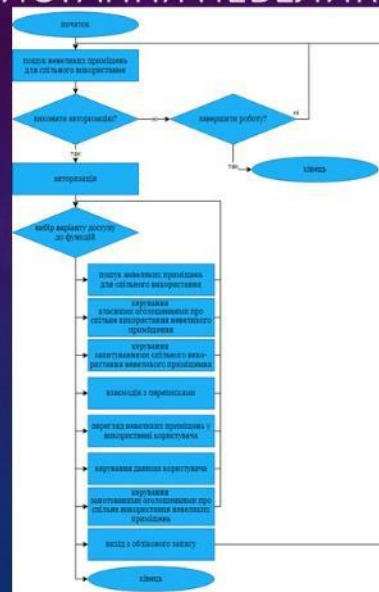


Рисунок В.8 – Схема бази даних програмного забезпечення спільного використання невеликих приміщень

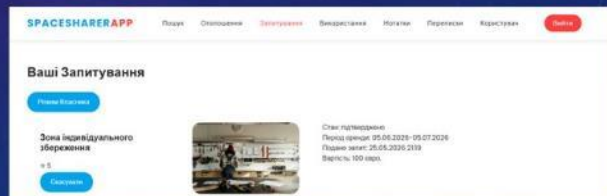
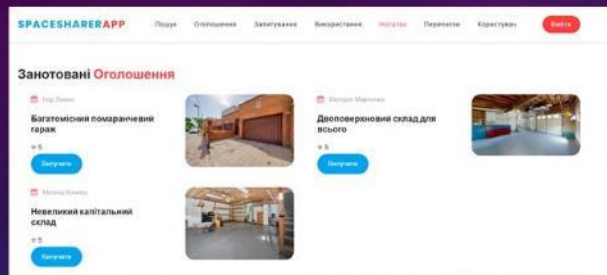
СХЕМА ФУНКЦІОНУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ



9

Рисунок В.9 – Схема функціонування програмного забезпечення спільного використання невеликих приміщень

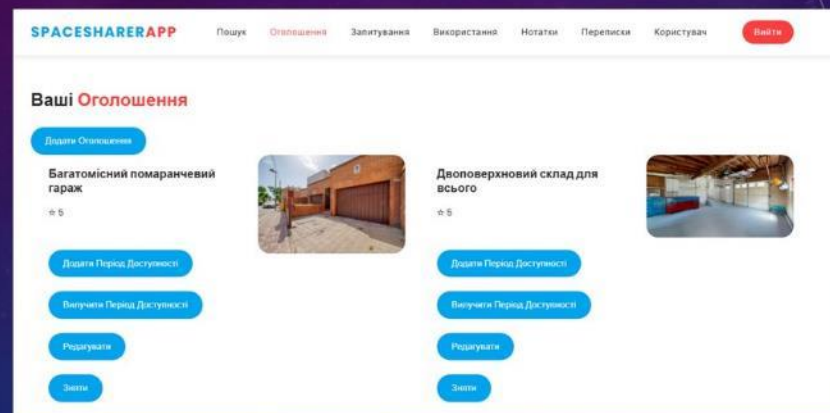
ІНТЕРФЕЙСИ ПРОГРАМИ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ ДЛЯ ОРЕНДАРЯ



10

Рисунок В.10 – Інтерфейси програми спільного використання невеликих приміщень для орендаря

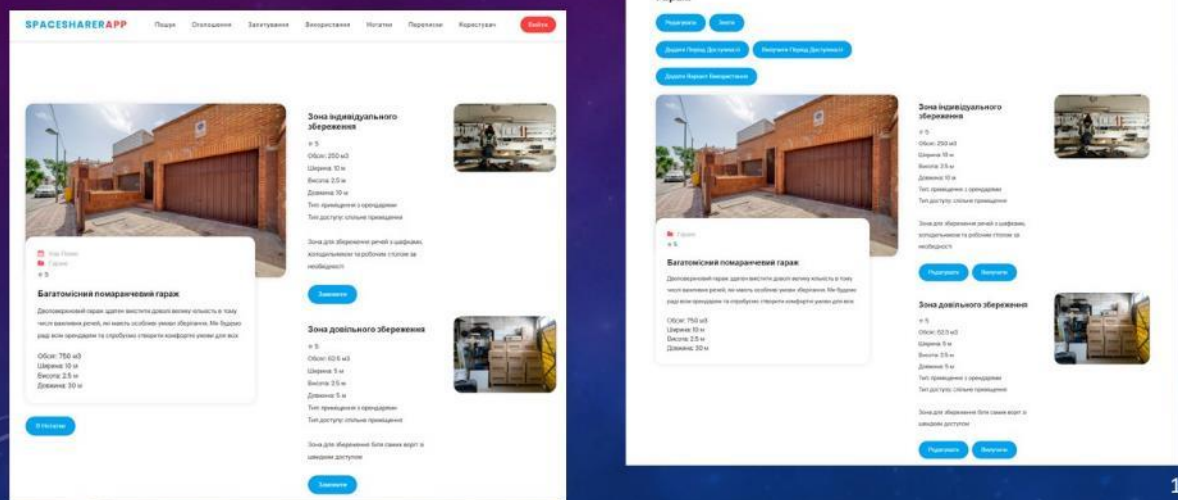
ІНТЕРФЕЙС ПРОГРАМИ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ ДЛЯ ВЛАСНИКА



11

Рисунок В.11 – Інтерфейс програми спільного використання невеликих приміщень для власника

ПЕРЕГЛЯД ОГЛОШЕННЯ ПРО СПІЛЬНЕ ВИКОРИСТАННЯ НЕВЕЛИКОГО ПРИМІЩЕННЯ



12

Рисунок В.12 – Перегляд оголошення про спільне використання невеликого приміщення

ТЕСТУВАННЯ ПРОГРАМИ СПІЛЬНОГО ВИКОРИСТАННЯ НЕВЕЛИКИХ ПРИМІЩЕНЬ

Функція	Результат	Функція	Результат	Функція	Результат
Додавання оголошення про спільне використання невеликого приміщення	+	Підтвердження запитування спільного використання невеликого приміщення	+	Відправлення повідомлення за невеликим приміщенням для спільного використання	+
Редагування оголошення про спільне використання невеликого приміщення	+	Закриття спільного використання невеликого приміщення	+	Перегляд повідомлень за невеликим приміщенням для спільного використання у орендаря	+
Зняття оголошення про спільне використання невеликого приміщення	+	Перегляд сторінки власника невеликих приміщень	+	Перегляд невеликих приміщень для спільного використання, за якими здійснювалась переписка	+
Перегляд власних оголошень про спільне використання невеликих приміщень	+	Перегляд сторінки орендаря спільного використання невеликих приміщень	+	Перегляд переписок за невеликим приміщенням для спільного використання у власника	+
Перегляд невеликих приміщень у використанні користувача	+	Заповнення користувацьких даних	+	Регістрація облікового запису	+
Перегляд оголошення про спільне використання невеликого приміщення	+	Перегляд оголошень про спільне використання невеликих приміщень власника	+	Авторизація	+
Нотування оголошення про спільне використання невеликого приміщення	+	Створення відгуку про досвід спільного використання невеликого приміщення	+	Додавання періоду доступності невеликого приміщення для спільного використання	+
Перегляд заповнення оголошень про спільне використання невеликих приміщень	+	Перегляд відгуків про досвід спільного використання за невеликим приміщенням	+	Видалення періоду доступності невеликого приміщення для спільного використання	+
Відміна нотування оголошення про спільне використання невеликого приміщення	+	Перегляд відгуків про досвід спільного використання за власником невеликих приміщень	+	Скасвання запитування спільного використання невеликого приміщення	+
Запитування спільного використання невеликого приміщення	+	Створення відгуку про досвід взаємодії з орендарем спільного використання невеликого приміщення	+	Перегляд скасованих запитувань спільного використання за всіма невеликими приміщеннями власника	+
Редгування варіанту спільного використання за невеликим приміщенням	+	Перегляд відгуків про досвід взаємодії за спільним використанням за орендарем	+	Перегляд всіх запитувань спільного використання невеликих приміщень за орендарем	+
Видалення варіанту спільного використання за невеликим приміщенням	+	Перегляд запитувань спільного використання за невеликим приміщенням	+	Пошук невеликих приміщень для спільного використання	+
		Перегляд запитувань спільного використання за всіма невеликими приміщеннями власника	+	Додавання варіанту спільного використання за невеликим приміщенням	+
		Перегляд підтверджених запитувань спільного використання за всіма невеликими приміщеннями власника	+		

13

Рисунок В.13 – Тестування програми спільного використання невеликих приміщень

ВИСНОВКИ

У виконаній роботі досягнуто зменшення витрат на оренду невеликих приміщень для зберігання речей шляхом розділення цих витрат між різними користувачами на основі розробки програмного забезпечення спільного використання невеликих приміщень. Для розділення витрат запропоновано кожному власнику визначати сукупність варіантів спільного використання невеликого приміщення, створюючи таким чином окремі комірки, частини, за допомогою яких відділяти простір для різних орендарів у складі одного невеликого приміщення.

Розроблена програма може використовуватись як власниками невеликих приміщень, так і потенційними або фактичними орендарями для того, щоб розподілити наявний простір у невеликих приміщеннях для оренди різних користувачів або запропонувати для спільного використання весь наявний простір невеликого приміщення, керувати запитом зацікавлених осіб у такій оренді, шукати варіанти, які влаштовують користувачів.

14

Рисунок В.14 – Висновки