

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 проекту (роботи)

бакалавра

(ступінь вищої освіти)

на тему ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ОРЕНДИ

(назва теми)

ОБЛАДНАННЯ ДЛЯ БУДІВЕЛЬНИХ РОБІТ. SOFTWARE FOR

CONSTRUCTION EQUIPMENT RENTAL

Виконав(ла): студент(ка) 4 курсу, групи КНТ-110

Спеціальності 121 Інженерія програмного

(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

ШЕПЛЯКОВ Г.К.

(ПРІЗВИЩЕ та ініціали)

Керівник СКРУПСЬКИЙ С.Ю.

(ПРІЗВИЩЕ та ініціали)

Рецензент ГОЛУБ Т.В.

(ПРІЗВИЩЕ та ініціали)

2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

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
« _____ » _____ 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ШЕПЛЯКОВА Георгія Костянтиновича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення оренди обладнання для будівельних робіт. Software for Construction Equipment Rental

керівник проєкту (роботи) к.т.н., доцент СКРУПСЬКИЙ Степан Юрійович _____,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від « 24 » квітня 2024 року № 168

2. Строк подання студентом проєкту (роботи) 10 червня 2024 року

3. Вихідні дані до проєкту (роботи) технічне завдання

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

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

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРІЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	СКРУПСЬКИЙ С.Ю., доцент		
Нормоконтроль	ДЕЙНЕГА Л.Ю., ст. викладач		

7. Дата видачі завдання « 15 » квітня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

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

Студент(ка)

(підпис) Георгій ШЕПЛЯКОВ
(Ім'я ПРІЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Степан СКРУПСЬКИЙ
(Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 91 с., 25 рис., 2 табл., 3 дод., 7 джерел.

ОРЕНДА ОБЛАДНАННЯ, БУДІВЕЛЬНІ РОБОТИ,
ВЕБЗАСТОСУНОК, БАЗА ДАНИХ, ФРЕЙМВОРК.

Об'єкт роботи – процес оренди обладнання для будівельних робіт.

Предмет роботи – програмне забезпечення оренди обладнання для будівельних робіт.

Мета роботи – розробити програмне забезпечення оренди обладнання для будівельних робіт, щоб забезпечити можливість взаємодії між різними орендодавцями та орендарями.

Матеріали, методи та технічні засоби: мова Python, система PostgreSQL, фреймворк Django.

Результати. Результатом є програма, призначена для реалізації послуг з оренди обладнання для будівельних робіт, що підтримує потенційно можливості і короткострокової, і довгострокової оренди від різних орендодавців, забезпечує опрацювання не тільки загальних моделей обладнання для будівельних робіт у наявності в орендодавця, але і конкретних екземплярів, для яких можуть вноситися дані про виявлені вади, що сприяє прийняттю обґрунтованих рішень орендарем.

Висновки. Виконана розробка дозволяє орендарям отримувати більшу кількість пропозицій за рахунок доступу до різних орендодавців одночасно, контролювати стан конкретних екземплярів обладнання, що сприяє зручності орендарів.

Галузь використання. Оренда будівельного обладнання на короткий та тривалий строк. Може використовуватися різними окремими компаніями-орендодавцями одночасно.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 91 pages, 25 figures, 2 tables, 3 appendixes, 7 sources.

EQUIPMENT RENTAL, CONSTRUCTION WORK, WEBSITE, DATABASE, FRAMEWORK.

The object of the work is the process of renting equipment for construction works.

The subject of the work is software for renting equipment for construction works.

The purpose of the work is to develop a software for renting equipment for construction works to ensure the possibility of interaction between different lessors and lessees.

Materials, methods and technical means: Python language, PostgreSQL system, Django framework.

The results. The result is a program designed for the implementation of construction equipment rental services. The software supports both short-term and long-term rental from various lessors, provides processing of not only general models of equipment for construction work available at the lessor, but also specific samples. Data on detected defects can be entered for these samples. It helps the lessees making informed decisions.

Conclusions. The completed development allows lessees to receive a larger number of offers due to access to different lessors at the same time, control the condition of specific instances of equipment, which contributes to the convenience of lessees.

Field of use. Rental of construction equipment for short and long term. It can be used by different individual rental companies at the same time.

ЗМІСТ

	С.
Перелік скорочень та умовних познач.....	7
Вступ.....	8
1 Аналіз предметної області.....	9
1.1 Аналіз застосунків оренди обладнання для будівельних робіт	9
1.2 Порівняльний аналіз розробки та аналогів	11
1.3 Висновки за розділом 1	13
2 Розробка архітектури програми.....	15
2.1 Вибір засобів проєктування	15
2.2 Сценарії використання програми	16
2.3 Структура бази даних	21
2.4 Висновки за розділом 2	31
3 Реалізація програми	32
3.1 Визначення порядку пошуку обладнання для будівельних робіт	32
3.2 Результати реалізації програмних одиниць додатку	33
3.3 Висновки за розділом 3	49
4 Експлуатація та тестування програми	50
4.1 Призначення програми	50
4.2 Умови виконання програми	50
4.3 Виконання і тестування програми.....	50
4.4 Висновки за розділом 4	55
Висновки	57
Перелік джерел посилання	58
Додаток А Технічне завдання	59
Додаток Б Текст програми	64
Додаток В Слайди презентації.....	85

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

MVT – Model-View-Template;

SQL – Structured Query Language;

БД – база даних.

ВСТУП

Обладнання для будівельних робіт є достатньо часто дороговартісним, а тому не завжди воно є у власності тих команд, які зайняті будівництвом. Це обумовлюється наступними конкретними причинами також.

Такі випадки виникають, коли стосуються специфічного обладнання, яке використовується не завжди, а також випадків, коли команда невелика, але для виконання конкретного ремонту, будівництва потребує певного великогабаритного обладнання.

Окрім того це стосується випадків, коли будівельні роботи виконуються самими власниками. Тоді з досить високою ймовірністю у власника щонайменше не все будівельне обладнання є в наявності.

Тож у всіх цих випадках необхідно забезпечити можливість оренди обладнання для будівельних робіт, а джерелом такого обладнання можуть бути різні компанії-орендодавці.

При створенні вебзастосунку для однієї конкретної такої компанії це значно звужує пошук, а створення вебзастосунку, який може бути використаний як певна платформа для оренди будівельного обладнання є перспективним практичним завданням. Окрім того варто проаналізувати можливості аналогів, які вже існують на ринку та доступні користувачам, щоб зрозуміти, які додаткові можливості можна надати в програмі, щоб підвищити зручність роботи орендарів зокрема, інформованість орендарів.

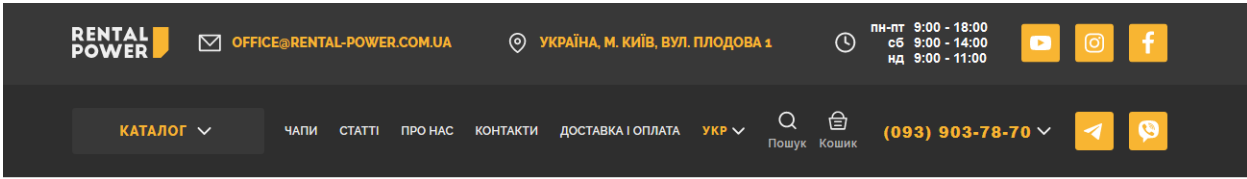
Відповідно мету роботи визначено як розробку програмного забезпечення оренди обладнання для будівельних робіт, щоб забезпечити можливість взаємодії між різними орендодавцями та орендарями.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз застосунків оренди обладнання для будівельних робіт

Результати аналізу застосунків оренди обладнання для будівельних робіт містять цілий ряд прикладів, з яких було обрано ті, що мають певні особливості.

Компанія Rental-Power [1] (рис. 1.1) більше спеціалізується на оренді енергетичного обладнання, однак це обладнання може використовуватися і для будівельних робіт. При визначенні терміну оренди задається режим до 12 годин або більше. Це дозволяє врахувати варіанти короткострокової оренди, однак вона фактично обмежена половиною доби. Тобто вибір можливого часу оренди є достатньо негнучким.



Rental Power / Оренда обладнання / Генератори / Генератор бензиновий HONDA EU 20 IS (1,5кВт)

Генератор бензиновий HONDA EU 20 IS (1,5кВт)

Є в наявності

Бренд: [Бензиновые генераторы Honda](#)

Термін оренди: 3 дд . мм . гггг по дд . мм . гггг Режим оренди: до 12 годин

від 400 грн з ПДВ

[ЗАМОВИТИ](#) [Замовити в один клік](#)

	1 - 4	5 - 14	15 - 30	31 *
грн/доба до 12 г/д	550	500	450	400
грн/доба від 12 г/д	715	650	585	520

Рисунок 1.1 – Вебзастосунок компанії Rental-Power

За кожною моделлю обладнання визначені фотографії, характеристики, коментарі орендарів. Доступ же до самого обладнання можна здійснити або

через каталог, або через пошук, але пошук містить тільки введення тексту (фактично назви) і більше ніяких параметрів. Каталог не дозволяє визначити період оренди, тобто щоб перевірити вартість оренди, потрібно зайти на сторінку обладнання спочатку.

Компанія United Rentals (рис. 1.2) є найбільшою компанією з оренди обладнання у світі, здійснює свою роботу в США та Канаді [2].

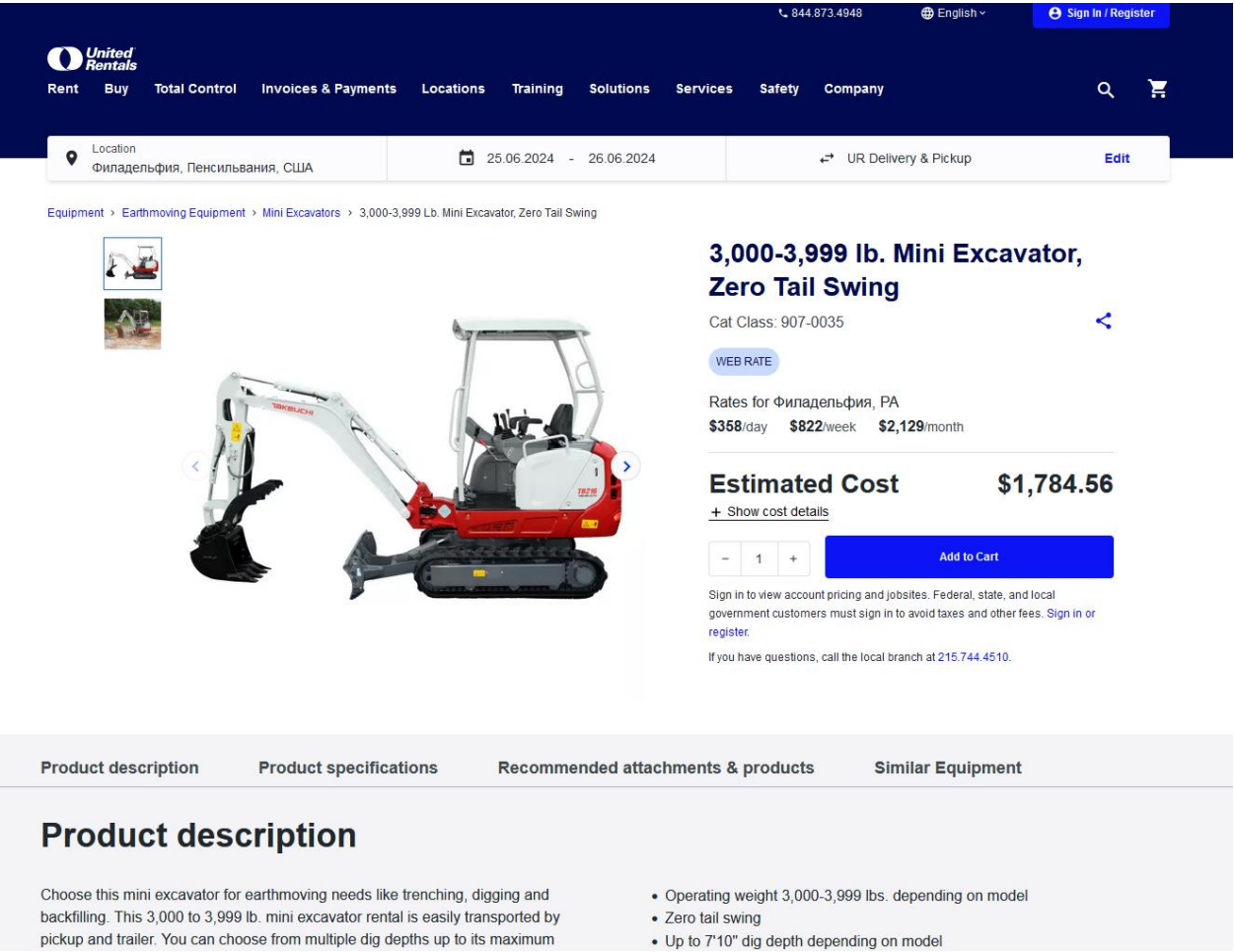


Рисунок 1.2 – Вебзастосунок компанії United Rentals

Встановлені періоди оренди, що визначають вартість, як день, тиждень або місяць. Тобто орендувати обладнання для будівельних робіт можна на певну кількість днів, місяців, тижнів. Під час перегляду сторінки обладнання потрібно задати місто, дати оренди, що впливає на результати. Якщо такого обладнання в місті немає на задані дати, то вартість не відображається. При цьому обидва застосунки дозволяють замовити оренду. Відгуки орендарями

не надаються. Пошук обладнання здійснюється з визначенням категорії обладнання.

Компанія BigRentz [3] (рис. 1.3) також є великим орендарем обладнання, але таким, який надає саме важке обладнання. Відмінністю даного вебзастосунок є те, що при відображенні обладнання за категорією можна задати властивості обладнання, таким чином фільтруючи результати. Але під час пошуку також можна вказати тільки назву. Вказати період оренди взагалі можна тільки під час формування замовлення, що загалом можна вважати певною незручністю для користувачів.

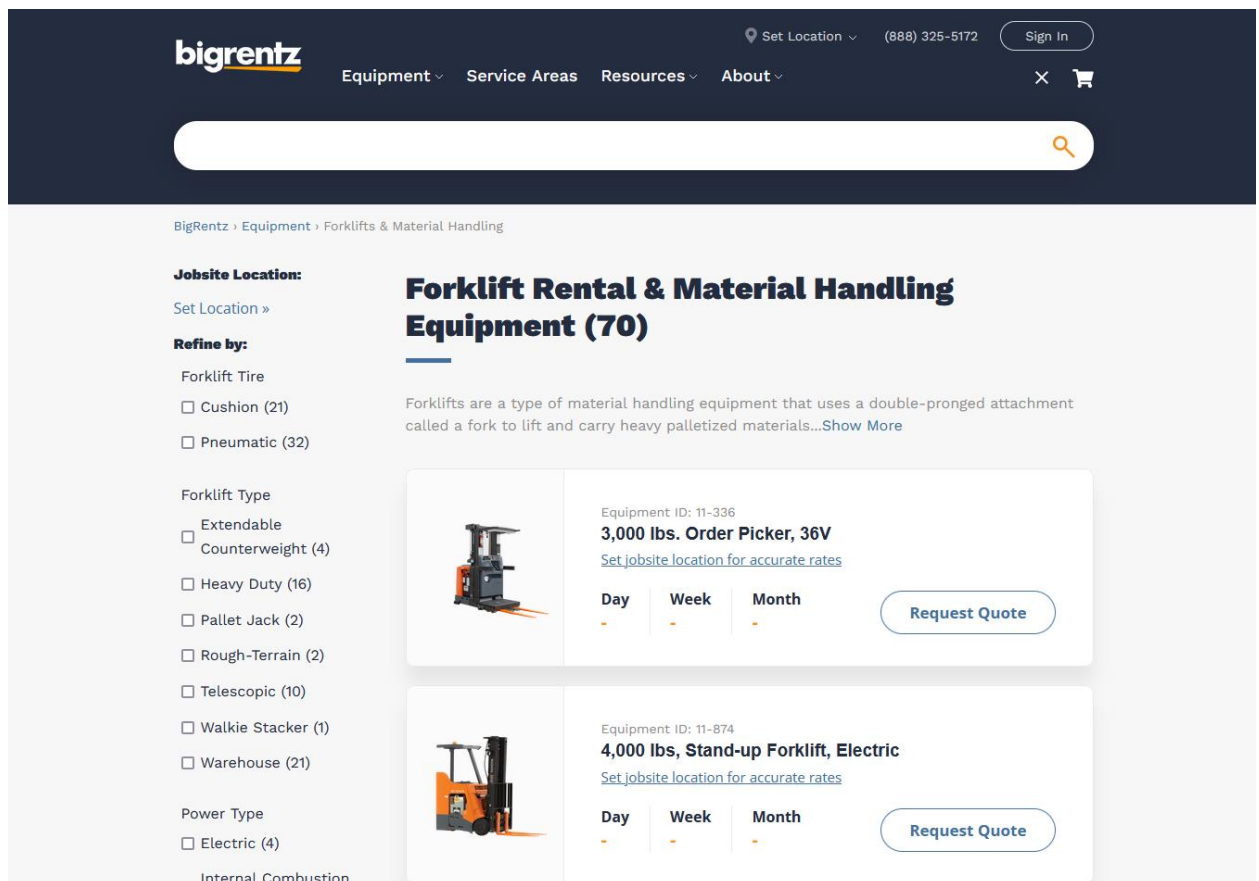


Рисунок 1.3 – Вебзастосунок компанії BigRentz

1.2 Порівняльний аналіз розробки та аналогів

Під час порівняння виділено вебзастосунки Rental-Power, United Rentals та виконувану розробку (табл. 1.1).

Таблиця 1.1 – Порівняння розробки програми оренди обладнання для будівельних робіт

Критерій	Rental-Power	United Rentals	Моя програма
Пошук обладнання для будівельних робіт	За назвою	За назвою, категорією та з визначенням міста, періоду оренди	За назвою, категорією, виробником, властивостями, з визначенням міста, періоду оренди, обмеженням оцінки, вартості
Джерело оренди	Компанія Rental-Power	Компанія United Rentals	Різні зареєстровані компанії-орендодавці
Період оренди	Півдоби або кількість діб	Кількість діб, тижнів, місяців	Кількість годин, діб, тижнів, місяців
Відгуки	Про обладнання	Відсутні	Про обладнання, про орендарів
Визначення вад екземплярів обладнання	Відсутнє	Відсутнє	Наявне

Порівняння вказує на те, що обидва застосунки-аналоги є вебсайтами окремих компаній, відповідно орендувати можна обладнання тільки цих компаній, в той час як розробка має об'єднати всіх, хто готовий надати в оренду власне обладнання, таким чином стаючи платформою для роботи різних орендодавців.

Розробка передбачає повноцінний пошук обладнання для будівельних робіт за назвою, категорією, виробником, властивостями, з визначенням міста, періоду оренди, обмеженням оцінки, вартості, в той час як у Rental-Power можна задати тільки назву, а у United Rentals відсутні властивості обладнання, виробник, неможливо обмежити оцінку, адже її і немає, і вартість оренди на заданий період.

Окрім того у розробці заплановано, що орендодавець сам має визначати можливості тривалості оренди, тому потенційно оренда може здійснюватися і на години. У United Rentals взагалі відсутні відгуки про обладнання, а у Rental-Power відгуки тільки про обладнання, у той час як в розробці враховуються відгуки як про обладнання, так і про орендарів.

У Rental-Power та United Rentals задаються тільки загальні моделі, конкретні екземпляри не представлені, тому зрозуміти, який саме екземпляр дістанеться в оренду, які в нього можуть бути проблеми, неможливо. А якщо компанія велика, то така інформація тим більш є важливою, адже це зворотній зв'язок від клієнтів, і можна не врахувати і не знати про такі вади взагалі.

Тож у підсумку виділено цілий ряд відмінностей розробки, що виконується, від аналогів.

1.3 Висновки за розділом 1

У розділі виконано аналіз програм оренди обладнання для будівельних робіт, включаючи вебзастосунки компаній Rental-Power, United Rentals, BigRentz, визначено відмінності розробки, що включають зокрема принцип оренди обладнання для будівельних робіт у різних орендодавців, можливість визначення вад екземплярів обладнання тощо.

Завдання на роботу:

- проаналізувати програми оренди обладнання для будівельних робіт;
- визначити засоби проєктування програми;

- виділити вимоги до програми оренди обладнання для будівельних робіт;
- виконати реалізацію програми оренди обладнання для будівельних робіт.

2 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМИ

2.1 Вибір засобів проєктування

PHP і Python розглядались в якості мов програмування для веброзробки як одні з найбільш використовуваних саме для цієї задачі, а подальше забезпечення технологіями розглядалось на основі порівняння відповідних фреймворків Symfony [4]-[5] та Django [6] (табл. 2.1). Не зважаючи на те, що Symfony використовує шаблон проєктування Model-View-Controller, у ньому реалізується підтримка шаблонів сторінок, що наближує це рішення до Model-View-Template (MVT) у Django. Django має додаткові засоби забезпечення захисту, в тому числі захисту від Structured Query Language (SQL)-ін'єкцій, коли у введеному користувачем тексті міститься SQL-запит.

Таблиця 2.1 – Вибір фреймворку для розробки програми оренди обладнання для будівельних робіт

Критерій	Symfony	Django
Підхід до побудови	Модульний	Монолітний
Мова	PHP	Python
Шаблон проєктування	Model-View-Controller, але з наявністю шаблонів	MVT
Рушій підтримки шаблонів	Twig	Django
Наявність підтримки	Менш розвинена	Більш розвинена
Засоби забезпечення безпеки	Аутентифікація, збереження паролів, контроль доступу за ролями	Аутентифікація, збереження паролів, контроль доступу за ролями, SQL-ін'єкції та багато інших

Для реалізації обрано фреймворк Django, оскільки його монолітність для даного проєкту є плюсом, адже дозволяє вести розробку, маючи потрібні засоби одразу, не витрачаючи час на побудову з компонентів базових засобів як у Symfony, до того ж це позитивно впливає на забезпечення безпеки, оскільки наявно багато засобів захисту. Також Django має кращу підтримку для навчання і вирішення проблем.

2.2 Сценарії використання програми

Неавторизований користувач має в основному потоці наступні сценарії:

- пошук обладнання для будівельних робіт для оренди;
- ознайомлення з моделлю обладнання для будівельних робіт, що включає сценарії визначення вартості оренди обладнання для будівельних робіт та перегляду доступних екземплярів моделі обладнання для будівельних робіт;
- надання моделей обладнання для будівельних робіт у орендодавця;
- ознайомлення з орендодавцем обладнання для будівельних робіт;
- ознайомлення з орендарем обладнання для будівельних робіт;
- надання відгуків про задану модель обладнання для будівельних робіт у орендодавця;
- надання відгуків про орендаря обладнання для будівельних робіт (рис. 2.1).

Користувач-орендар обладнання для будівельних робіт має в основному потоці наступні сценарії:

- замовлення обладнання для будівельних робіт в оренду;
- визначення виявленої вади екземпляра обладнання для будівельних робіт у орендодавця;
- підтримання виявленої вади екземпляра обладнання для будівельних робіт у орендодавця;

- редагування даних орендаря обладнання для будівельних робіт;
- публікація відгуку про модель обладнання для будівельних робіт у орендодавця;
- надання відгуків орендаря обладнання для будівельних робіт;
- формування власного переліку обладнання для будівельних робіт;
- додавання моделі обладнання для будівельних робіт у орендодавця до власного переліку;
- надання власного переліку обладнання для будівельних робіт.
- авторизація;
- надання власних замовлень обладнання для будівельних робіт в оренду.

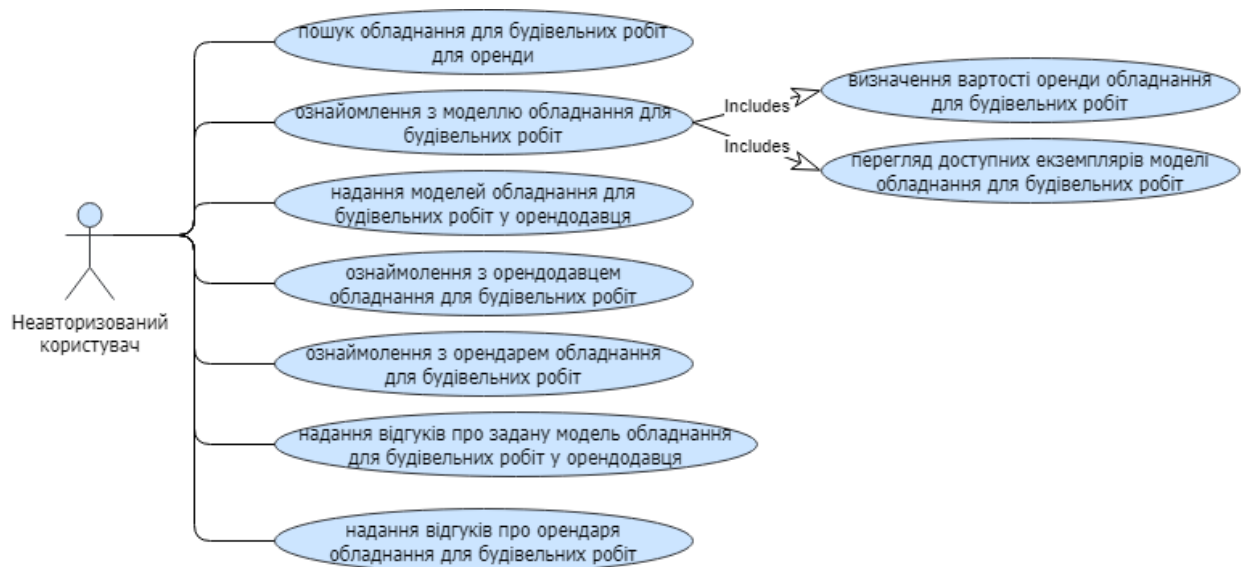


Рисунок 2.1 – Діаграма прецедентів програми оренди обладнання для будівельних робіт для неавторизованого користувача

Альтернативні сценарії користувача-орендаря обладнання для будівельних робіт:

- спростування виявленої вади екземпляра обладнання для будівельних робіт у орендодавця, що є альтернативним для підтримання виявленої вади екземпляра обладнання для будівельних робіт у орендодавця, коли наступний

орендар визначив, що інформація, внесена попереднім орендарем є некоректною;

– реєстрація є альтернативною для авторизації, коли користувач-орендар обладнання для будівельних робіт не має облікового запису;

– редагування власного переліку обладнання для будівельних робіт, що є альтернативним для формування власного переліку обладнання для будівельних робіт, коли перелік має назву, яка має бути оновлена;

– вилучення моделі обладнання для будівельних робіт у орендодавця з власного переліку, що є альтернативним для додавання моделі обладнання для будівельних робіт у орендодавця до власного переліку, коли модель вже не цікавить орендаря (рис. 2.2).

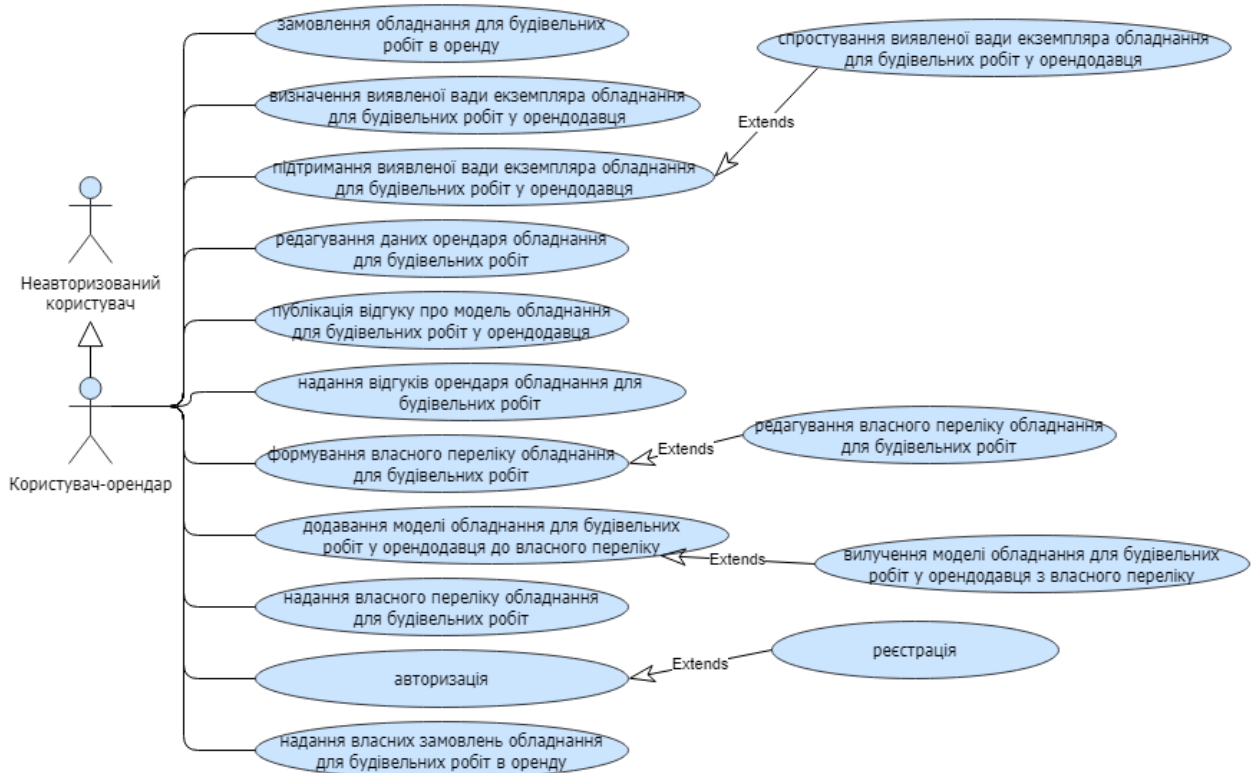


Рисунок 2.2 – Діаграма прецедентів програми оренди обладнання для будівельних робіт для користувача-орендаря обладнання для будівельних робіт

Користувач-працівник орендодавця обладнання для будівельних робіт

має в основному потоці наступні сценарії:

- додавання моделі обладнання для будівельних робіт у орендодавця;
- додавання екземпляра моделі обладнання для будівельних робіт у орендодавця;
- редагування даних орендодавця обладнання для будівельних робіт;
- публікація відгуку про орендаря обладнання для будівельних робіт;
- надання відгуків орендодавця обладнання для будівельних робіт;
- надання працівників орендодавця обладнання для будівельних робіт;
- авторизація;
- надання замовлень обладнання для будівельних робіт в оренду у орендодавця;
- надання замовлень обладнання для будівельних робіт в оренду у орендодавця конкретним орендарем;
- керування замовленням обладнання для будівельних робіт в оренду;
- додавання базової вартості оренди моделі обладнання для будівельних робіт у орендодавця.

Альтернативні сценарії користувача-працівника орендодавця обладнання для будівельних робіт:

- редагування моделі обладнання для будівельних робіт у орендодавця, що є альтернативним для додавання моделі обладнання для будівельних робіт у орендодавця, коли дані мають бути оновлені стосовно загальної моделі;
- редагування екземпляра моделі обладнання для будівельних робіт у орендодавця, що є альтернативним для додавання екземпляра моделі обладнання для будівельних робіт у орендодавця, коли дані мають бути оновлені стосовно екземпляра моделі обладнання для будівельних робіт;
- виведення з оренди екземпляра моделі обладнання для будівельних робіт, що є альтернативним для додавання екземпляра моделі обладнання для будівельних робіт у орендодавця, коли екземпляр моделі більше не може використовуватися для оренди;

- редагування базової вартості оренди моделі обладнання для будівельних робіт у орендодавця, що є альтернативним для додавання базової вартості оренди моделі обладнання для будівельних робіт у орендодавця, коли вартість змінилась, але зміни не можуть бути додані для роботи з певної дати, а мають бути внесені негайно;
- реєстрація користувача-працівника іншим користувачем-працівником, що є альтернативним для авторизації, коли з'являється новий працівник;
- реєстрація орендодавця обладнання для будівельних робіт, що є альтернативним для авторизації, коли з'являється не тільки додатковий працівник, а перший працівник орендодавця (рис. 2.3).



Рисунок 2.3 – Діаграма прецедентів програми оренди обладнання для будівельних робіт для користувача-працівника орендодавця обладнання для будівельних робіт

2.3 Структура бази даних

Реляційна база даних (БД) побудована для збереження даних у програмі оренди обладнання для будівельних робіт на основі системи PostgreSQL, що на даний момент є найпотужнішою відповідною системою, яка до того ж є безкоштовною у використанні.

Типи обладнання для будівельних робіт визначені в таблиці БД Constructequiptype зі стовпцями:

- id – ідентифікатор типу обладнання для будівельних робіт;
- equipmenttype_caption – назва типу обладнання для будівельних робіт.

З типами обладнання для будівельних робіт співставляються властивості, які дозволяють виконувати потім пошук потрібного обладнання. Самі значення властивостей співставляються з конкретним обладнанням для будівельних робіт, а ось властивості відповідають типам обладнання.

Властивості типів обладнання для будівельних робіт визначені в таблиці БД Constructequiptypeproperty зі стовпцями:

- id – ідентифікатор властивості типу обладнання для будівельних робіт;
- constructequiptype – тип обладнання для будівельних робіт, якому встановлена ця властивість: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiptype;
- property_caption – назва властивості, яка співвіднесена з цим типом обладнання для будівельних робіт.

Виробники обладнання для будівельних робіт визначені в таблиці БД Constructequipproucer зі стовпцями:

- id – ідентифікатор виробника обладнання для будівельних робіт;
- producer_name – назва компанії-виробника обладнання для будівельних робіт;

– `headquarters_country` – країна, в якій знаходиться офіс компанії-виробника обладнання для будівельних робіт.

Обладнання для будівельних робіт розділено на моделі такого обладнання та конкретні екземпляри, які вже орендуються конкретними людьми. Модель у даному випадку розглядається не як загальне поняття у виробника, а як модель, представлена вже у конкретного орендодавця.

Моделі обладнання для будівельних робіт у орендодавця визначені в таблиці БД `Constructequipment` зі стовпцями:

– `id` – ідентифікатор моделі обладнання для будівельних робіт у орендодавця;

– `constructequipproducer` – виробник моделі обладнання для будівельних робіт у орендодавця: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequipproducer`;

– `constructequiplessor` – орендодавець, який володіє цією моделлю обладнання для будівельних робіт: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiplessor`;

– `constructequiptype` – тип обладнання для будівельних робіт, якому відповідає ця модель: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiptype`;

– `equipment_model` – назва моделі обладнання для будівельних робіт у орендодавця;

– `equipment_presentation` – текстова презентація моделі обладнання для будівельних робіт у орендодавця;

– `equipment_photo` – основна фотографія моделі обладнання для будівельних робіт у орендодавця;

– `average_opinion` – оцінка, яку орендарі поставили цій моделі обладнання для будівельних робіт у орендодавця всередньому.

Поза основною фотографією моделі обладнання для будівельних робіт у орендодавця можуть мати і ряд додаткових фотографій, визначених в таблиці БД `Constructequipmentphoto` зі стовпцями:

- id – ідентифікатор додаткової фотографії моделі обладнання для будівельних робіт у орендодавця;

- constructequipment – модель обладнання для будівельних робіт, що сфотографована на додатковій фотографії: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequipment;

- equipment_photo – фотографія моделі обладнання для будівельних робіт у орендодавця.

Екземпляри моделей обладнання для будівельних робіт у орендодавця визначені в таблиці БД Constructequipmentsample зі стовпцями:

- id – ідентифікатор екземпляра моделі обладнання для будівельних робіт у орендодавця;

- constructequipment – модель обладнання для будівельних робіт у орендодавця, яка має цей екземпляр: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequipment;

- serialnumber – серійний номер екземпляра моделі обладнання для будівельних робіт у орендодавця;

- production_year – рік, коли конкретно цей екземпляр обладнання для будівельних робіт виготовлено;

- producer_country – країна, в якій було виготовлено конкретно цей екземпляр обладнання для будівельних робіт;

- equipment_photo – фотографія, на якій сфотографовано конкретно цей екземпляр обладнання для будівельних робіт (за необхідності та наявності);

- state_presentation – представлення текстом стану, в якому конкретно цей екземпляр обладнання для будівельних робіт є;

- active_rent – доступність або недоступність конкретно цього екземпляра обладнання для будівельних робіт для оренди.

Орендарі обладнання будівельних робіт, якщо вони зауважили, який саме екземпляр вони орендували, можуть описати ті вади, які має цей екземпляр.

Вади екземпляра обладнання для будівельних робіт визначені в таблиці БД Constructequipmentsampleproblem зі стовпцями:

- id – ідентифікатор вади екземпляра обладнання для будівельних робіт;
- constructequipmentsample – екземпляр вади екземпляра обладнання для будівельних робіт, виявленої користувачем-орендарем: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequipmentsample;
- constructequiprenter – користувач-орендар, який орендував цей екземпляр обладнання для будівельних робіт і зауважив виявлену ваду: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiprenter;
- problem_fixed – вада, яку було виділено користувачем-орендарем у екземпляра обладнання для будівельних робіт;
- problem_date – дата, коли було виявлено цим користувачем-орендарем цю ваду у екземпляра обладнання для будівельних робіт;
- critical_level – визначений користувачем-орендарем рівень критичності цієї вади для використання обладнання для будівельних робіт.

Підтвердження або спростування думки іншого користувача-орендаря про ваду у екземпляра обладнання для будівельних робіт визначені в таблиці БД Constructequipmentsampleproblemapprove зі стовпцями:

- id – ідентифікатор підтвердження або спростування думки іншого користувача-орендаря про ваду у екземпляра обладнання для будівельних робіт;
- constructequipmentsampleproblem – вада, яку було виділено користувачем-орендарем у екземпляра обладнання для будівельних робіт і на яку реагує цей користувач-орендар: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequipmentsampleproblem;
- constructequiprenter – користувач-орендар, який висловлює цим записом свою думку (підтвердження або спростування) щодо вади, яку в даного екземпляра обладнання для будівельних робіт виявив інший

користувач-орендар: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiprenter;

- approve_cancel – ця думка є підтвердженням або спростуванням думки щодо вади в цього екземпляра обладнання для будівельних робіт.

Значення властивостей моделі обладнання для будівельних робіт у орендодавця визначені в таблиці БД Constructequipproperty зі стовпцями:

- id – ідентифікатор значення властивості моделі обладнання для будівельних робіт у орендодавця;

- constructequiptypeproperty – властивість типу обладнання для будівельних робіт у орендодавця, якій визначено це значення: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiptypeproperty;

- propertyvalue – значення, яке визначено моделі обладнання для будівельних робіт у орендодавця;

- constructequipment – модель обладнання для будівельних робіт у орендодавця, якій відповідає значення властивості.

Користувачі-орендарі обладнання для будівельних робіт визначені в таблиці БД Constructequiprenter зі стовпцями:

- id – ідентифікатор користувача-орендаря обладнання для будівельних робіт;

- account – обліковий запис користувача-орендаря обладнання для будівельних робіт: цим зовнішнім ключем реалізується зв'язок з таблицею User, що стандартно забезпечується у БД через Django;

- renter_name – ім'я людини, яка є користувачем-орендарем обладнання для будівельних робіт;

- renter_phone – номер телефону для зв'язку користувача-орендаря обладнання для будівельних робіт;

- renter_photo – фотографія людини, яка є користувачем-орендарем обладнання для будівельних робіт;

- renter_description – опис себе користувачем-орендарем обладнання для будівельних робіт;

– `average_opinion` – оцінка, в яку було всередньому оцінено цього орендаря орендодавцями.

Оцінювання орендодавцями обладнання для будівельних робіт орендарів після взаємодії визначені в таблиці БД `Constructequiprenteropinion` зі стовпцями:

– `id` – ідентифікатор оцінювання орендодавцями обладнання для будівельних робіт орендарів після взаємодії;

– `constructequiprenter` – орендар обладнання для будівельних робіт, якого оцінив орендодавець після взаємодії: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiprenter`;

– `constructequiplessor` – орендодавець обладнання для будівельних робіт, який оцінив орендаря після взаємодії: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiplessor`;

– `opinion_level` – оцінка, що була поставлена;

– `opinion_text` – текст виставленої оцінки з загальним враженням орендодавцем обладнання для будівельних робіт орендаря після взаємодії;

– `opinion_date` – дата, коли оцінка внесена в програму.

Оцінювання орендарями моделей обладнання для будівельних робіт у орендодавців визначені в таблиці БД `Constructequipopinion` зі стовпцями:

– `id` – ідентифікатор оцінювання орендарем моделі обладнання для будівельних робіт у орендодавця;

– `constructequipment` – модель обладнання для будівельних робіт у орендодавця, яку оцінив орендар: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequipment`;

– `constructequiprenter` – користувач-орендар, який оцінив цю модель обладнання для будівельних робіт: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiprenter`;

– `opinion_level` – оцінка, що була поставлена користувачем-орендарем цій моделі обладнання для будівельних робіт;

– `opinion_text` – текст виставленої оцінки з загальним враженням орендарем про модель обладнання для будівельних робіт у орендодавця;

– `opinion_date` – дата, коли оцінка орендаря моделі обладнання для будівельних робіт у орендодавця внесена в програму.

Компанії-орендодавці обладнання для будівельних робіт визначені в таблиці БД `Constructequiplessor` зі стовпцями:

– `id` – ідентифікатор компанії-орендодавця обладнання для будівельних робіт;

– `lessor_caption` – назва компанії-орендодавця обладнання для будівельних робіт;

– `lessor_phone` – номер телефону компанії-орендодавця обладнання для будівельних робіт;

– `lessor_photo` – фотографія, якою проілюстровано компанію-орендодавця обладнання для будівельних робіт;

– `lessor_site` – вебсайт компанії-орендодавця обладнання для будівельних робіт;

– `lessor_description` – текстовий опис компанії-орендодавця обладнання для будівельних робіт;

– `function_city` – місто, в якому надає компанія-орендодавець своє обладнання для будівельних робіт в оренду;

– `average_opinion` – оцінка, в яку було всередньому оцінено все обладнання для будівельних робіт цього орендодавця орендарями.

Компанії-орендодавці не самі працюють з програмою, а через власних працівників. Користувачі-працівники компаній-орендодавців визначені в таблиці БД `Constructequiplessoruser` зі стовпцями:

– `id` – ідентифікатор користувача-працівника компанії-орендодавця обладнання для будівельних робіт;

– `constructequiplessor` – компанія-орендодавець, яку представляє цей користувач-працівник: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiplessor`;

– account – обліковий запис користувача-працівника компанії-орендодавця обладнання для будівельних робіт: цим зовнішнім ключем реалізується зв'язок з таблицею User;

– user_name – ім'я людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт;

– user_phone – номер телефону людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт;

– user_photo – фотографія людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт;

– position – посада (роль) людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт.

Резервування оренди будівельного обладнання визначені в таблиці БД Constructequiprental зі стовпцями:

– id – ідентифікатор резервування оренди будівельного обладнання;

– constructequiplessor – орендодавець, у якого виконано резервування оренди будівельного обладнання: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiplessor;

– constructequiprenter – орендар, який зарезервував оренду будівельного обладнання: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiprenter;

– rental_start – дата початку оренди будівельного обладнання;

– rental_end – дата, в яку має закінчитися оренда будівельного обладнання;

– rental_date – дата, коли резервування оренди будівельного обладнання відбулось;

– rental_price – вартість резервування оренди будівельного обладнання;

– rental_state – стан резервування оренди будівельного обладнання;

– rental_todo – інформаційне повідомлення від орендаря до орендодавця щодо резервування оренди будівельного обладнання.

Обладнання для будівельних робіт, що входить в резервування оренди, визначено в таблиці БД Constructequiprentalelement зі стовпцями:

- id – ідентифікатор обладнання для будівельних робіт, що входить в резервування оренди;
- constructequiprental – резервування оренди обладнання для будівельних робіт, за яким було цей екземпляр зарезервовано: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequiprental;
- constructequipmentsample – екземпляр моделі обладнання для будівельних робіт у орендодавця, який орендар вніс у своє резервування: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequipmentsample;
- fixsample – конкретно цей екземпляр моделі був вибраний випадковим чином у програмі або він був цілеспрямовано обраний серед всіх екземплярів користувачем-орендарем.

Вартість оренди обладнання для будівельних робіт за одиницю часу визначено в таблиці БД Constructequiprate зі стовпцями:

- id – ідентифікатор вартості оренди обладнання для будівельних робіт за одиницю часу;
- constructequipment – модель обладнання для будівельних робіт, за якою ця вартість оренди визначена: цим зовнішнім ключем реалізується зв'язок з таблицею Constructequipment;
- min_rate – кількість одиниць часу оренди, після якої може бути застосовано цю вартість при обчисленні вартості резервування;
- price_rate – вартість у гривнях одиниці часу оренди цієї моделі обладнання для будівельних робіт;
- price_unit – одиниця часу, яка визначена для цієї вартості моделі обладнання для будівельних робіт;
- active_date – дата, з якої ця вартість може використовуватися.

Користувачі-орендарі, готуючись до резервувань оренди, можуть створювати свої власні переліки обладнання для будівельних робіт. Такі

- `id` – ідентифікатор моделі обладнання для будівельних робіт, яка входить до переліку користувача-орендаря;
- `constructequiplist` – перелік обладнання для будівельних робіт користувача-орендаря, в який входить ця модель: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequiplist`;
- `constructequipment` – модель обладнання для будівельних робіт у орендодавця, яку внесено в цей перелік користувач-орендаря: цим зовнішнім ключем реалізується зв'язок з таблицею `Constructequipment`.

2.4 Висновки за розділом 2

Використання вебфреймворку Django разом з мовою Python має спростити процес розробки за рахунок використання вбудованих рішень для забезпечення елементарних можливостей і концентрації на розробці найбільш особливих можливостей. Визначення конкретних можливостей, які програма для оренди обладнання для будівельних робіт має підтримувати, здійснено. Структуру БД для збереження даних визначено і забезпечено використанням системи PostgreSQL.

3 РЕАЛІЗАЦІЯ ПРОГРАМИ

3.1 Визначення порядку пошуку обладнання для будівельних робіт

Пошук обладнання для будівельних робіт здійснюється так, що спочатку використовуються ті параметри, які стосуються моделей, реалізуючи відфільтровування тих моделей, які не підходять під ці параметри (рис. 3.1).

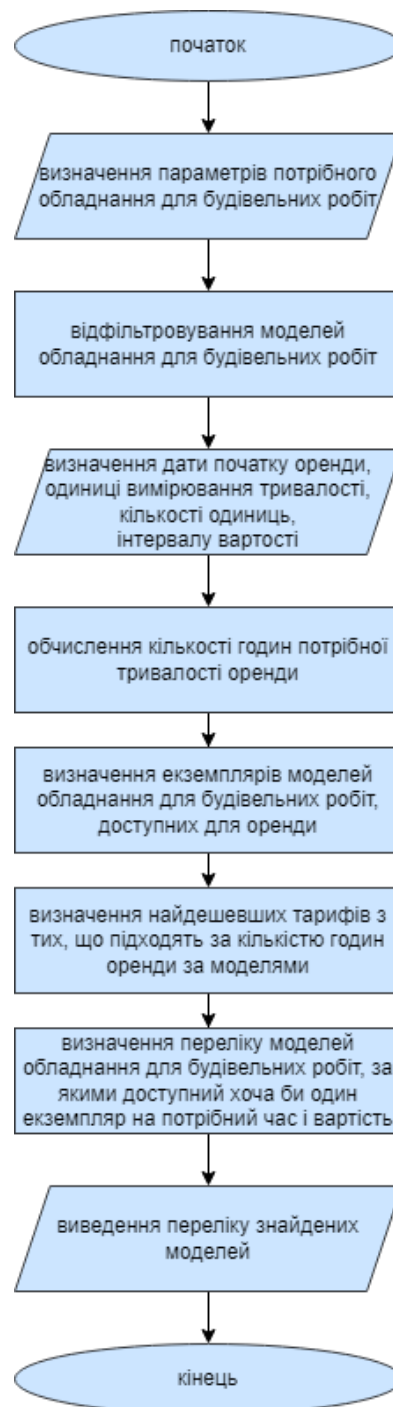


Рисунок 3.1 – Схема пошуку обладнання для будівельних робіт для оренди

Далі визначаються параметри самої оренди, однак ці параметри визначаються в певних одиницях, тобто тривалість оренди може визначатися або в кількості годин, або діб, або тижнів, або місяців. У свою чергу тарифи на вартість оренди моделей обладнання орендодавцем також приводяться в різних одиницях. Тому може виявитися, що одиниці не співпадають. Тому спочатку визначається тривалість потрібної оренди в годинах, а потім вона співставляється з мінімальною тривалістю оренди в тарифах, відкидаючи ті, які не підходять, та вибираючи з тих, що залишились, найдорожчий. Тоді після співставлення визначаються ті моделі обладнання для будівельних робіт, за якими є хоча би один екземпляр, який доступний на потрібні дати оренди, а вартість його оренди підходить до заданих параметрів.

3.2 Результати реалізації програмних одиниць додатку

При реалізації вебзастосунку застосовано фреймворк Django: структура реалізована з забезпеченням необхідних представлень, які взаємодіють з шаблонами вебсторінок, реалізованими на основі безкоштовного шаблону [7], та моделями, що сумарно реалізує шаблон проєктування MVT.

Функція представлення `constructequipmentview` реалізує ознайомлення з моделлю обладнання для будівельних робіт, а одночасно і визначення вартості оренди обладнання для будівельних робіт, перегляд доступних екземплярів моделі обладнання для будівельних робіт, які входять до відповідного прецедента. Для цього функція має такі параметри:

- `id_constructequipment` – номер моделі обладнання для будівельних робіт у орендаря;

- `constrental_start` – додатковий параметр, який визначає час і дату початку оренди цієї моделі обладнання для будівельних робіт у орендаря;

– `constrental_unit` – додатковий параметр, який визначає одиниці виміру тривалості оренди цієї моделі обладнання для будівельних робіт у орендаря: 0 – година, 1 – доба, 2 – тиждень, 3 – місяць;

– `constrental_duration` – додатковий параметр, який визначає кількість одиниць виміру тривалості оренди цієї моделі обладнання для будівельних робіт у орендаря.

Оскільки ці 3 останні параметри є додатковими, тобто можуть не визначатися під час виклику, але визначення вартості оренди входить у відповідну логіку роботи, то у випадку, якщо значення не були задані, то автоматично враховується оренда на наступний день (рівня через добу) на 1 годину.

Дана функція визначає всі дані щодо цієї моделі обладнання для будівельних робіт, а якщо зазначено невірний номер, то повертається сторінка з повідомленням про помилку та можливими причинами. До даних входять поля самої моделі, моделі орендодавця, значень властивостей, фотографій, входження у переліки, створені авторизованим користувачем-орендарем, дані екземплярів цієї моделі обладнання для будівельних робіт у орендодавця разом з заявленими вадами цих екземплярів.

Для визначення вартості оренди цієї моделі обладнання для будівельних робіт у орендодавця спочатку реалізується перевірка наявності замовлень на потрібні дати, визначені параметрами. Якщо таких замовлень, виключаючи ті, які ще не є підтвердженими або є скасованими, немає, то обчислюється вартість замовлення оренди.

Функція представлення `searchconstructequipmentview` реалізує пошук обладнання для будівельних робіт для оренди. Пошук здійснюється за вказаними фільтрами, а отримані в підсумку моделі обладнання для будівельних робіт впорядковуються за оцінкою, що всередньому виставляли орендарі. Параметри пошуку обладнання для будівельних робіт можуть включати такі:

– виробник обладнання для будівельних робіт;

- тип обладнання для будівельних робіт;
- назва моделі обладнання для будівельних робіт (може вводитися неповністю);
- потрібні значення властивостей моделі обладнання для будівельних робіт, що відповідають обраному типу обладнання;
- мінімальна оцінку, яку повинна мати модель обладнання для будівельних робіт;
- дата і час початку оренди обладнання для будівельних робіт;
- одиниця вимірювання часу оренди обладнання для будівельних робіт;
- потрібна тривалість оренди обладнання для будівельних робіт в одиницях вимірювання, які задано;
- максимально допустима вартість оренди обладнання для будівельних робіт;
- мінімально прийнятна вартість оренди обладнання для будівельних робіт.

За результатами пошуку виключаються ті моделі обладнання для будівельних робіт, які не мають жодних вільних екземплярів для оренди на вказаний період часу, а також ті, вартість оренди яких не входить у вказаний проміжок.

Замовлення обладнання для будівельних робіт в оренду реалізується двома функціями представлень: `constructequipmentrentalgetview` і `constructequipmentrentalpostview`.

Функція `constructequipmentrentalgetview` працює за методом GET і дозволяє обробити сформований набір моделей або екземплярів моделей обладнання для будівельних робіт, визначаючи вартість, формуючи відповідно пропозицію замовлення. Оскільки вона може працювати і з моделями, і з екземплярами, то якщо задано моделі, то випадково вибирається один з доступних для оренди на вказані дату і час (а також тривалість такої оренди) екземплярів. Тоді за кожним обраним користувачем екземпляром або

за кожним екземпляром, обраним випадково за заданою моделлю, визначається вартість оренди, а також сумарна вартість.

Функція `constructequipmentrentalpostview` вже безпосередньо відповідає за створення замовлення обладнання для будівельних робіт, тобто вона відповідає за підтвердження такого замовлення за сформованою пропозицією. Після виконання цієї дії, якщо задано коректні параметри замовлення, то замовлення разом зі своїм складом з екземплярів обладнання для будівельних робіт зберігається в БД.

Функція представлення `lessorconstructequipmentsviiew` реалізує надання моделей обладнання для будівельних робіт у орендодавця, номер якого задається параметром `id_lessor`. Вибираються всі моделі обладнання для будівельних робіт, які наявні у заданого орендаря для оренди.

Функція представлення `constructlessorview` реалізує ознайомлення з орендодавцем обладнання для будівельних робіт, номер якого задається параметром `id_lessor`. Вибираються тільки 9 моделей обладнання будівельних робіт, що наявні для оренди у цього орендодавця, 9 відгуків про нього від орендарів та значення базових полів моделі.

Функція представлення `addconstructequipmentview` реалізує додавання моделі обладнання для будівельних робіт у орендодавця, що працює за допомогою форм, реалізованих класами `ConstructequipmentForm` та `ConstructequippropertyForm` відповідно для самої моделі та пов'язаних властивостей. Аналогічно влаштована функція представлення `editconstructequipmentview` для редагування моделі обладнання для будівельних робіт у орендодавця, але яка має параметр `id_constructequipment`, що визначає номер моделі обладнання для будівельних робіт у орендодавця, але при цьому застосовується перевірка, що користувач авторизувався як працівник саме цього орендодавця.

Функція представлення `addconstructequipmentsampleproblemview` реалізує визначення виявленої вади екземпляра обладнання для будівельних робіт у орендодавця на основі форми `ConstructequipmentsampleproblemForm` та

заданого значення параметра `id_constructequipmentsample`, що задає номер екземпляра моделі обладнання для будівельних робіт у орендодавця. У функції здійснюється перевірка, що цей користувач орендував цей екземпляр моделі обладнання для будівельних робіт у цього орендодавця.

Функція представлення `constructequipmentsampleproblemapproveview` реалізує підтримання виявленої вади екземпляра обладнання для будівельних робіт у орендодавця, де вада задається параметром `id_constructequipmentsampleproblem`. Так само функцією `constructequipmentsampleproblemcancelview` реалізовано спростування.

Функція представлення `addconstructequipmentsampleview` реалізує додавання екземпляра моделі обладнання для будівельних робіт у орендодавця, що працює за допомогою форми, реалізованої класом `ConstructequipmentsampleForm`. Збереження можна виконати тільки, якщо авторизований користувач представляє того самого орендодавця, якому належить відповідна модель, яка задається параметром `id_constructequipment`. Аналогічно влаштована функція представлення `editconstructequipmentsampleview` для редагування екземпляра моделі обладнання для будівельних робіт у орендодавця, але яка має параметр `id_constructequipmentsample`, що визначає номер екземпляра моделі обладнання для будівельних робіт у орендодавця, але при цьому застосовується перевірка, що користувач авторизувався як працівник саме цього орендодавця.

Функція представлення `deactivateconstructequipmentsampleview` реалізує виведення з оренди екземпляра моделі обладнання для будівельних робіт з номером `id_constructequipmentsample`, але при цьому застосовується перевірка, що користувач авторизувався як працівник саме цього орендодавця, який володіє цим екземпляром. Усі пов'язані майбутні замовлення програмно скасовуються.

Функція представлення `constructequiprenterview` реалізує ознайомлення з орендарем обладнання для будівельних робіт з номером

`id_constructequiprenter`. Сюди входять відгуки самого орендаря обладнання для будівельних робіт, відгуки про нього, основна інформація. Якщо номер орендаря не задано, то береться з даних авторизації.

Функція представлення `addconstructequiprenteropinionview` реалізує публікацію відгуку про орендаря обладнання для будівельних робіт з номером `id_constructequiprenter` на основі форми класу `ConstructequiprenteropinionForm`, перевіряючи, що цей орендар орендував обладнання для будівельних робіт у цього орендодавця.

Функція представлення `addconstructequiporinionview` реалізує публікацію відгуку про модель обладнання для будівельних робіт у орендодавця з номером `id_constructequipment` на основі форми класу `ConstructequiporinionForm`, перевіряючи, що цей орендар орендував таку модель обладнання для будівельних робіт у цього орендодавця.

Функція представлення `constructequiporinionsview` реалізує надання відгуків про задану модель обладнання для будівельних робіт у орендодавця з номером `id_constructequipment`.

Функція представлення `constructequiprenteropinionsview` реалізує надання відгуків про орендаря обладнання для будівельних робіт з номером `id_constructequiprenter`.

Функція представлення `constructequiprenteropinionsview` реалізує надання відгуків орендаря обладнання для будівельних робіт з номером `id_constructequiprenter`.

Функція представлення `constructequiplessoropinionsview` реалізує надання відгуків орендодавця обладнання для будівельних робіт з номером `id_constructequiplessor`.

Функція представлення `constructequiplessorusersview` реалізує надання працівників орендодавця обладнання для будівельних робіт, вибираючи їх за тим орендодавцем, працівником якого є авторизований користувач.

Функція представлення `lessorconstructequiprentalsview` реалізує надання замовлень обладнання для будівельних робіт в оренду у орендодавця

з параметрами `rental_state`, що визначає стан замовлень обладнання для будівельних робіт, `sort`, що визначає напрям сортування, та `rental_start`, `rental_end`, що задають проміжок дат замовлень для відображення. Замовлення оренди обладнання для будівельних робіт визначаються за орендодавцем, працівником якого є авторизований користувач.

Функція представлення `renterconstructequiprentalsview` реалізує надання власних замовлень обладнання для будівельних робіт в оренду з параметрами `rental_state`, що визначає стан замовлень обладнання для будівельних робіт, `sort`, що визначає напрям сортування, та `rental_start`, `rental_end`, що задають проміжок дат замовлень для відображення. Замовлення оренди обладнання для будівельних робіт визначаються за орендарем, який створив їх і який має бути авторизований у програмі на момент звернення.

Функція представлення `renterlessorconstructequiprentalsview` реалізує надання замовлень обладнання для будівельних робіт в оренду у орендодавця конкретним орендарем з номером `id_constructequiprenter`, параметри ж орендодавця визначаються за тим, працівником якого роботодавця є авторизований користувач.

Функція представлення `manageconstructequiprentalview` реалізує керування замовленням обладнання для будівельних робіт в оренду з номером `id_constructequiprental`, змінюючи його стан, якщо така зміна можлива, на стан `rental_state`. Дії виконуються, якщо це замовлення стосується того орендодавця, працівником якого є авторизований користувач.

Функція представлення `addconstructequiprateview` реалізує додавання базової вартості оренди моделі обладнання для будівельних робіт у орендодавця на основі класу форми `ConstructequiprateForm`. Параметр `id_constructequipment` визначає конкретну модель обладнання для будівельних робіт, для якої це треба зробити. Через цю форму також працює функція `editconstructequiprateview` для редагування базової вартості оренди моделі обладнання для будівельних робіт у орендодавця з номером `id_constructequiprate`.

Функція представлення `addconstructequiplistview` реалізує формування власного переліку обладнання для будівельних робіт з назвою, що визначається параметром `equipmentlistname`.

Функція представлення `editconstructequiplistview` реалізує редагування власного переліку обладнання для будівельних робіт з номером `id_constructequiplist`, який має належати авторизованому користувачу.

Функція представлення `addconstructequiplistelementview` реалізує додавання моделі обладнання для будівельних робіт у орендодавця з номером `id_constructequipment` до власного переліку `id_constructequiplist`, який має належати авторизованому користувачу.

Функція представлення `deleteconstructequiplistelementview` реалізує вилучення моделі обладнання для будівельних робіт у орендодавця з номером `id_constructequipment` з власного переліку `id_constructequiplist`, який має належати авторизованому користувачу.

Функція представлення `constructequiplistview` реалізує надання власного переліку обладнання для будівельних робіт за номером `id_constructequiplist`, перевіряючи, що переліком володіє авторизований користувач-орендар.

Модель типів обладнання для будівельних робіт реалізована класом `Constructequiptype` з одним полем `equipmenttype_caption` – поле назви типу обладнання для будівельних робіт (тип `CharField`).

Модель властивостей типів обладнання для будівельних робіт реалізована класом `Constructequiptypeproperty` з полями:

- `constructequiptype` – поле типу обладнання для будівельних робіт, якому встановлена ця властивість, що зв'язуванням з моделлю `Constructequiptype` реалізовано типом `ForeignKey`;

- `property_caption` – поле назви властивості, яка співвіднесена з цим типом обладнання для будівельних робіт (тип `CharField`).

Модель виробників обладнання для будівельних робіт реалізована класом `Constructequipproducer` з полями:

- `producer_name` – поле назви компанії-виробника обладнання для будівельних робіт (тип `CharField`);

- `headquarters_country` – поле країни, в якій знаходиться офіс компанії-виробника обладнання для будівельних робіт (тип `CharField`).

Моделі обладнання для будівельних робіт у орендодавця реалізована класом `Constructequipment` з полями:

- `constructequipproducer` – поле виробника моделі обладнання для будівельних робіт у орендодавця, що зв'язуванням з моделлю `Constructequipproducer` реалізовано типом `ForeignKey`;

- `constructequiplessor` – поле орендодавця, який володіє цією моделлю обладнання для будівельних робіт, що зв'язуванням з моделлю `Constructequiplessor` реалізовано типом `ForeignKey`;

- `constructequiptype` – поле типу обладнання для будівельних робіт, якому відповідає ця модель, що зв'язуванням з моделлю `Constructequiptype` реалізовано типом `ForeignKey`;

- `equipment_model` – поле назви моделі обладнання для будівельних робіт у орендодавця (тип `CharField`);

- `equipment_presentation` – поле текстової презентації моделі обладнання для будівельних робіт у орендодавця (тип `CharField`);

- `equipment_photo` – поле основної фотографії моделі обладнання для будівельних робіт у орендодавця (тип `ImageField`);

- `average_opinion` – поле оцінки, яку орендарі поставили цій моделі обладнання для будівельних робіт у орендодавця всередньому (тип `FloatField`).

Модель додаткових фотографій моделі обладнання для будівельних робіт реалізована класом `Constructequipmentphoto` з полями:

- `constructequipment` – поле моделі обладнання для будівельних робіт, що сфотографована на додатковій фотографії, що зв'язуванням з моделлю `Constructequipment` реалізовано типом `ForeignKey`;

- `equipment_photo` – поле фотографії моделі обладнання для будівельних робіт у орендодавця (тип `ImageField`).

Модель екземпляра моделі обладнання для будівельних робіт у орендодавця реалізована класом `Constructequipmentsample` з полями:

- `constructequipment` – поле моделі обладнання для будівельних робіт у орендодавця, яка має цей екземпляр, що зв'язуванням з моделлю `Constructequipment` реалізовано типом `ForeignKey`;
- `serialnumber` – поле серійного номеру екземпляра моделі обладнання для будівельних робіт у орендодавця (тип `CharField`);
- `production_year` – поле року, коли конкретно цей екземпляр обладнання для будівельних робіт виготовлено (тип `PositiveSmallIntegerField`);
- `producer_country` – поле країни, в якій було виготовлено конкретно цей екземпляр обладнання для будівельних робіт (тип `CharField`);
- `equipment_photo` – поле фотографії, на якій сфотографовано конкретно цей екземпляр обладнання для будівельних робіт (тип `ImageField`);
- `state_presentation` – поле представлення текстом стану, в якому конкретно цей екземпляр обладнання для будівельних робіт є (тип `PositiveSmallIntegerField`);
- `active_rent` – поле доступності або недоступності конкретно цього екземпляра обладнання для будівельних робіт для оренди (тип `BooleanField`).

Модель вади екземпляра обладнання для будівельних робіт реалізована класом `Constructequipmentsampleproblem` з полями:

- `constructequipmentsample` – поле екземпляру вади екземпляра обладнання для будівельних робіт, виявленої користувачем-орендарем, що зв'язуванням з моделлю `Constructequipmentsample` реалізовано типом `ForeignKey`;
- `constructequiprenter` – поле користувача-орендаря, який орендував цей екземпляр обладнання для будівельних робіт і зауважив виявлену ваду, що зв'язуванням з моделлю `Constructequiprenter` реалізовано типом `ForeignKey`;
- `problem_fixed` – поле вади, яку було виділено користувачем-орендарем у екземпляра обладнання для будівельних робіт (тип `CharField`);

– `problem_date` – поле дати, коли було виявлено цим користувачем-орендарем цю ваду у екземпляра обладнання для будівельних робіт (тип `DateField`);

– `critical_level` – поле визначеного користувачем-орендарем рівня критичності цієї вади для використання обладнання для будівельних робіт (тип `PositiveSmallIntegerField`).

Модель підтвердження або спростування думки іншого користувача-орендаря про ваду у екземпляра обладнання для будівельних робіт реалізована класом `Constructequipmentsampleproblemapprove` з полями:

– `constructequipmentsampleproblem` – поле вади, яку було виділено користувачем-орендарем у екземпляра обладнання для будівельних робіт і на яку реагує цей користувач-орендар, що зв'язуванням з моделлю `Constructequipmentsampleproblem` реалізовано типом `ForeignKey`;

– `constructequiprenter` – поле користувача-орендаря, який висловлює цим записом свою думку (підтвердження або спростування) щодо вади, яку в даного екземпляра обладнання для будівельних робіт виявив інший користувач-орендар, що зв'язуванням з моделлю `Constructequiprenter` реалізовано типом `ForeignKey`;

– `approve_cancel` – поле визначення того, що думка є підтвердженням або спростуванням думки щодо вади в цього екземпляра обладнання для будівельних робіт (тип `BooleanField`).

Модель значення властивостей моделі обладнання для будівельних робіт у орендодавця реалізована класом `Constructequiptypeproperty` з полями:

– `constructequiptypeproperty` – поле властивості моделі обладнання для будівельних робіт у орендодавця, якій визначено це значення, що зв'язуванням з моделлю `Constructequiptypeproperty` реалізовано типом `ForeignKey`;

– `propertyvalue` – поле значення, яке визначено моделі обладнання для будівельних робіт у орендодавця (тип `CharField`);

– `constructequipment` – поле моделі обладнання для будівельних робіт у орендодавця, якій відповідає значення властивості, що зв'язуванням з моделлю `Constructequipment` реалізовано типом `ForeignKey`.

Модель користувача-орендаря обладнання для будівельних робіт реалізована класом `Constructequipmentrenter` з полями:

– `account` – поле облікового запису користувача-орендаря обладнання для будівельних робіт, що зв'язуванням з моделлю `User` реалізовано типом `OneToOneField`;

– `renter_name` – поле імені людини, яка є користувачем-орендарем обладнання для будівельних робіт (тип `CharField`);

– `renter_phone` – поле номеру телефону для зв'язку користувача-орендаря обладнання для будівельних робіт (тип `CharField`);

– `renter_photo` – поле фотографії людини, яка є користувачем-орендарем обладнання для будівельних робіт (тип `ImageField`);

– `renter_description` – поле опису себе користувачем-орендарем обладнання для будівельних робіт (тип `CharField`);

– `average_opinion` – поле оцінки, в яку було всередньому оцінено цього орендаря орендодавцями (тип `FloatField`).

Модель оцінювання орендодавцями обладнання для будівельних робіт орендарів після взаємодії реалізована класом `Constructequipmentrenteropinion` з полями:

– `constructequipmentrenter` – поле орендаря обладнання для будівельних робіт, якого оцінив орендодавець після взаємодії, що зв'язуванням з моделлю `Constructequipmentrenter` реалізовано типом `ForeignKey`;

– `constructequipmentlessor` – поле орендодавця обладнання для будівельних робіт, який оцінив орендаря після взаємодії, що зв'язуванням з моделлю `Constructequipmentlessor` реалізовано типом `ForeignKey`;

– `opinion_level` – поле оцінки, що була поставлена (тип `PositiveSmallIntegerField`);

– `opinion_text` – поле тексту виставленої оцінки з загальним враженням орендодавцем обладнання для будівельних робіт орендаря після взаємодії (тип `CharField`);

– `opinion_date` – поле дати, коли оцінка внесена в програму (тип `DateTimeField`).

Модель оцінювання орендарями моделей обладнання для будівельних робіт у орендодавців реалізована класом `Constructequipmentopinion` з полями:

– `constructequipment` – поле моделі обладнання для будівельних робіт у орендодавця, яку оцінив орендар, що зв'язуванням з моделлю `Constructequipment` реалізовано типом `ForeignKey`;

– `constructequipmentrenter` – поле користувача-орендаря, який оцінив цю модель обладнання для будівельних робіт, що зв'язуванням з моделлю `Constructequipmentrenter` реалізовано типом `ForeignKey`;

– `opinion_level` – поле оцінки, що була поставлена користувачем-орендарем цій моделі обладнання для будівельних робіт (тип `PositiveIntegerField`);

– `opinion_text` – поле тексту виставленої оцінки з загальним враженням орендарем про модель обладнання для будівельних робіт у орендодавця (тип `CharField`);

– `opinion_date` – поле дати, коли оцінка орендаря моделі обладнання для будівельних робіт у орендодавця внесена в програму (тип `DateTimeField`).

Модель компанії-орендодавця обладнання для будівельних робіт реалізована класом `Constructequipmentlessor` з полями:

– `lessor_caption` – поле назви компанії-орендодавця обладнання для будівельних робіт (тип `CharField`);

– `lessor_phone` – поле номеру телефону компанії-орендодавця обладнання для будівельних робіт (тип `CharField`);

– `lessor_photo` – поле фотографії, якою проілюстровано компанію-орендодавця обладнання для будівельних робіт (тип `ImageField`);

– `lessor_site` – поле вебсайту компанії-орендодавця обладнання для будівельних робіт (тип `URLField`);

– `lessor_description` – поле текстового опису компанії-орендодавця обладнання для будівельних робіт (тип `CharField`);

– `function_city` – поле міста, в якому надає компанія-орендодавець своє обладнання для будівельних робіт в оренду (тип `CharField`);

– `average_opinion` – поле оцінки, в яку було всередньому оцінено все обладнання для будівельних робіт цього орендодавця орендарями (тип `FloatField`).

Модель користувача-працівника компанії-орендодавця реалізована класом `Constructequiplessoruser` з полями:

– `constructequiplessor` – поле компанії-орендодавця, яку представляє цей користувач-працівник, що зв'язуванням з моделлю `Constructequiplessor` реалізовано типом `ForeignKey`;

– `account` – поле облікового запису користувача-працівника компанії-орендодавця обладнання для будівельних робіт, що зв'язуванням з моделлю `User` реалізовано типом `OneToOneField`;

– `user_name` – поле імені людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт (тип `CharField`);

– `user_phone` – поле номеру телефону людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт (тип `CharField`);

– `user_photo` – поле фотографії людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт (тип `ImageField`);

– `position` – поле посади людини, яка є користувачем-працівником компанії-орендодавця обладнання для будівельних робіт (тип `CharField`).

Модель резервування оренди будівельного обладнання реалізована класом `Constructequiprental` з полями:

– `constructequiplessor` – поле орендодавця, у якого виконано резервування оренди будівельного обладнання, що зв'язуванням з моделлю `Constructequiplessor` реалізовано типом `ForeignKey`;

– `constructequiprenter` – поле орендаря, який зарезервував оренду будівельного обладнання, що зв'язуванням з моделлю `Constructequiprenter` реалізовано типом `ForeignKey`;

– `rental_price` – поле вартості резервування оренди будівельного обладнання (тип `FloatField`);

– `rental_start` – поле дати початку оренди будівельного обладнання (тип `DateTimeField`);

– `rental_end` – поле дати, в яку має закінчитися оренда будівельного обладнання (тип `DateTimeField`);

– `rental_date` – поле дати, коли резервування оренди будівельного обладнання відбулось (тип `DateTimeField`);

– `rental_state` – поле стану резервування оренди будівельного обладнання (тип `PositiveSmallIntegerField`);

– `rental_todo` – поле інформаційного повідомлення від орендаря до орендодавця щодо резервування оренди будівельного обладнання (тип `CharField`).

Модель обладнання для будівельних робіт, що входить в резервування оренди, реалізована класом `Constructequiprentalelement` з полями:

– `constructequiprental` – поле резервування оренди обладнання для будівельних робіт, за яким було цей екземпляр зарезервовано, що зв'язуванням з моделлю `Constructequiprental` реалізовано типом `ForeignKey`;

– `constructequipmentsample` – поле екземпляра моделі обладнання для будівельних робіт у орендодавця, який орендар вніс у своє резервування, що зв'язуванням з моделлю `Constructequipmentsample` реалізовано типом `ForeignKey`;

– `fixsample` – поле визначення того, що конкретно цей екземпляр моделі був вибраний випадковим чином у програмі або він був цілеспрямовано обраний серед всіх екземплярів користувачем-орендарем (тип `BooleanField`).

Модель вартості оренди обладнання для будівельних робіт за одиницю часу реалізована класом `Constructequirate` з полями:

– `constructequipment` – поле моделі обладнання для будівельних робіт, за якою ця вартість оренди визначена, що зв'язуванням з моделлю `Constructequipment` реалізовано типом `ForeignKey`;

– `min_rate` – поле кількості одиниць часу оренди, після якої може бути застосовано цю вартість при обчисленні вартості резервування (тип `PositiveSmallIntegerField`);

– `price_rate` – поле вартості у гривнях одиниці часу оренди цієї моделі обладнання для будівельних робіт (тип `FloatField`);

– `price_unit` – поле одиниці часу, яка визначена для цієї вартості моделі обладнання для будівельних робіт (тип `PositiveSmallIntegerField`);

– `active_date` – поле дати, з якої ця вартість може використовуватися (тип `DateField`).

Модель переліку обладнання для будівельних робіт реалізована класом `Constructequiplist` з полями:

– `equipmentlist_name` – поле назви переліку обладнання для будівельних робіт користувача-орендаря (тип `CharField`);

– `constructequiprenter` – поле користувача-орендаря, якому належить цей перелік обладнання для будівельних робіт, що зв'язуванням з моделлю `Constructequiprenter` реалізовано типом `ForeignKey`;

– `constructequipments` – поле переліку моделей обладнання для будівельних робіт у орендодавця, які збережено користувачем-орендарем, що зв'язуванням з моделлю `Constructequipment` реалізовано типом `ManyToManyField`.

3.3 Висновки за розділом 3

За шаблоном MVT реалізовано всю функціональність програми з забезпеченням всіх необхідних складових та описом представлень і моделей вебзастосунку.

4 ЕКСПЛУАТАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМИ

4.1 Призначення програми

Програма призначена для реалізації послуг з оренди обладнання для будівельних робіт, підтримуючи потенційно можливості і короткострокової, і довгострокової оренди від різних орендодавців, забезпечуючи опрацювання не тільки загальних моделей обладнання для будівельних робіт у наявності в орендодавця, але і конкретних екземплярів, для яких можуть вноситися дані про виявлені вади, що сприяє прийняттю обґрунтованих рішень орендарем.

4.2 Умови виконання програми

Обов'язковою умовою експлуатації програми оренди обладнання для будівельних робіт є наявність адміністратора програми або користувача, здатного виконувати такі функції. В обов'язки адміністратора входить наповнення даними: типами обладнання для будівельних робіт, виробниками, властивостями. Також адміністратор повинен виконати розгортання додатку на сервері.

Технічне забезпечення сервера повинно включати мінімально: процесор Intel Xeon E5440 (частота 2,83 ГГц), жорсткий диск – 100 Гб (наявність фотографій може значно збільшувати вимоги до сервера, тому це потребує контролю та можливо введення обмежень на обсяг фотографій, які користувач може завантажувати, інакше необхідно розширювати потрібний простір), оперативна пам'ять – 4 Гб.

4.3 Виконання і тестування програми

Виконання програми змінюється в залежності від того, який саме користувач, тобто в якій саме ролі, з нею працює. Оскільки кожна роль має свій набір функцій, то було проведено тестування і перевірено в залежності від

кожної ролі, чи працює кожна потрібна функція та яким чином. Тестування виявилось успішним, адже в підсумку всі результати відповідають тим, які очікувались.

Неавторизований користувач має дуже просте меню, що включає тільки пункт для пошуку обладнання для будівельних робіт за встановленими параметрами:

- виробник обладнання для будівельних робіт;
- тип обладнання для будівельних робіт;
- назва моделі обладнання для будівельних робіт (може вводиться неповністю);
- потрібні значення властивостей моделі обладнання для будівельних робіт, що відповідають обраному типу обладнання;
- мінімальна оцінку, яку повинна мати модель обладнання для будівельних робіт;
- дата і час початку оренди обладнання для будівельних робіт;
- одиниця вимірювання часу оренди обладнання для будівельних робіт;
- потрібна тривалість оренди обладнання для будівельних робіт в одиницях вимірювання, які задано;
- максимально та мінімальна допустима вартість оренди обладнання для будівельних робіт.

З результатів можна відкрити потрібне обладнання для будівельних робіт на окремій сторінці (рис. 4.1). Ця сторінка включає інформацію про цю модель обладнання для будівельних робіт, відгуки, екземпляри, доступні для оренди у цього орендодавця. Неавторизований користувач може також авторизуватись і якщо він є орендодавцем, то він зможе додавати до власних переліків ті моделі, які переглядає (рис. 4.2). Створювати нові переліки можна з пункту меню переліків, де відображаються всі власні переліки користувача та кнопка додавання нового.

Модель будівельного обладнання

Дриль-шурупверт
DCD991P2

Трьохшвидкісний дриль-шурупверт з акумулятором від відомого виробника. Є представником останнього покоління компанії. Зручний у використанні і отримує дуже позитивні відгуки від майстрів.

-  DeWalt
компанія-виробник, США
- ✓ Три
Кількість швидкостей
- ✓ 13
Діаметр патрона, мм
- ✓ Літій-іонний
Акумулятор
- ✓ 95
Максимальний крутний момент, Нм
- ✓ 820
Потужність
- ✓ Два
Вага, кг




5.0
оцінка орендарів


ОрендуЙКо
орендодавець

Рисунок 4.1 – Вебсторінка моделі обладнання для будівельних робіт у орендодавця у відображенні неавторизованого користувача

Модель будівельного обладнання

Дриль-шурупверт
DCD991P2

У переліках: робочий інструмент (вилучити)

Додати до переліку

Додати

Трьохшвидкісний дриль-шурупверт з акумулятором від відомого виробника. Є представником останнього покоління компанії. Зручний у використанні і отримує дуже позитивні відгуки від майстрів.

-  DeWalt
компанія-виробник, США
- ✓ Три
Кількість швидкостей
- ✓ 13
Діаметр патрона, мм
- ✓ Літій-іонний
Акумулятор
- ✓ 95
Максимальний крутний момент, Нм
- ✓ 820
Потужність




5.0
оцінка орендарів


ОрендуЙКо
орендодавець

Рисунок 4.2 – Вебсторінка моделі обладнання для будівельних робіт у орендодавця у відображенні користувача-орендаря

На сторінці обладнання для будівельних робіт доступна інформація про екземпляри моделі у цього орендодавця (рис. 4.3). За кожним екземпляром відображається інформація про рік виготовлення, країну виробництва, опис стану від орендодавця та зафіксовані орендарями, які орендували цей екземпляр, вади. Можна додати нову ваду, але внесення відбудеться тільки, якщо через програму раніше цей орендар орендував саме цей екземпляр обладнання. Зверху на картці екземпляра демонструється серійний номер. Кожний запис про ваду можна або підтримати, або спростувати за допомогою відповідно стрілочок вгору або вниз, натискаючи на них. Кількість таких підтверджень та спростувань показується біля стрілок.

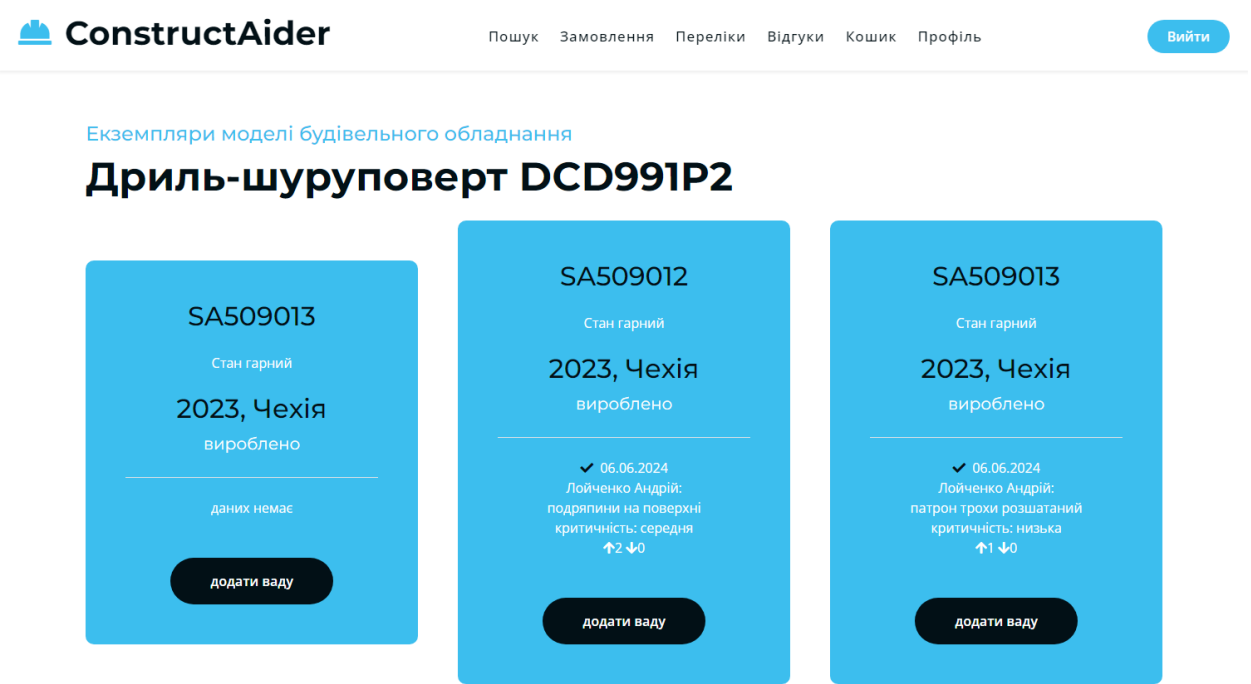


Рисунок 4.3 – Екземпляри моделі обладнання для будівельних робіт

На тій же сторінці моделі обладнання для будівельних робіт можна перевірити доступність моделі обладнання, вартість та доступні для замовлення для оренди екземпляри (рис. 4.4). Можна або замовити саму модель, тоді екземпляр додається випадково, або вибрати конкретний екземпляр. Кнопка замовити додає модель або екземпляр обладнання для будівельних робіт до кошика. Щоб завершити замовлення, потрібно обрати з

меню пункт з кошиком. Після перевірки параметрів замовлення необхідно перейти до підтвердження замовлення. Тільки тоді воно створюється і відображається у орендодавця.

Переглянути ж всі власні замовлення обладнання для будівельних робіт можна через меню замовлень. Меню відгуків дозволяє переглянути власні відгуки.

The screenshot displays the ConstructAider web interface. At the top, there is a navigation bar with the logo and menu items: Пошук, Замовлення, Переліки, Відгуки, Кошик, Профіль, and a Вийти button. The main content area is divided into two columns. The left column contains a 'Параметри оренди' (Rental Parameters) form with fields for 'Початок оренди: дата' (Start date), 'час' (time), 'Тривалість оренди: одиниця' (Duration unit), and 'кількість' (quantity). Below these fields is a 'Перевірити' (Check) button. The right column displays a list of equipment models for rent. The first item is 'Оренда обладнання моделі DeWalt DCD991P2 (Дриль-шурупверт)' with a price of '200.00 €' and a 'Замовити' (Order) button. Below it are two more items, 'SA509012' and 'SA509013', each with a 'Замовити' button. Each item also has a link to 'Екземпляр моделі відобразити вади' (Display model example defects).

Рисунок 4.4 – Перевірка параметрів оренди моделі обладнання для будівельних робіт

Працівник орендодавця можуть керувати власним обладнанням компанії, визначаючи яке саме доступне для оренди, додаючи, знімаючи навпаки його з оренди, роблячи це все через меню обладнання. Також можна додавати (реєструвати) працівників та переглядати поточних через меню працівників.

З меню замовлень можна відобразити всі замовлення до цього орендодавця (рис. 4.5). Можна задати потрібний стан або відобразити всі,

варіант сортування, а також проміжок дат замовлення обладнання для будівельних робіт. Якщо період не задано, то відображаються всі.

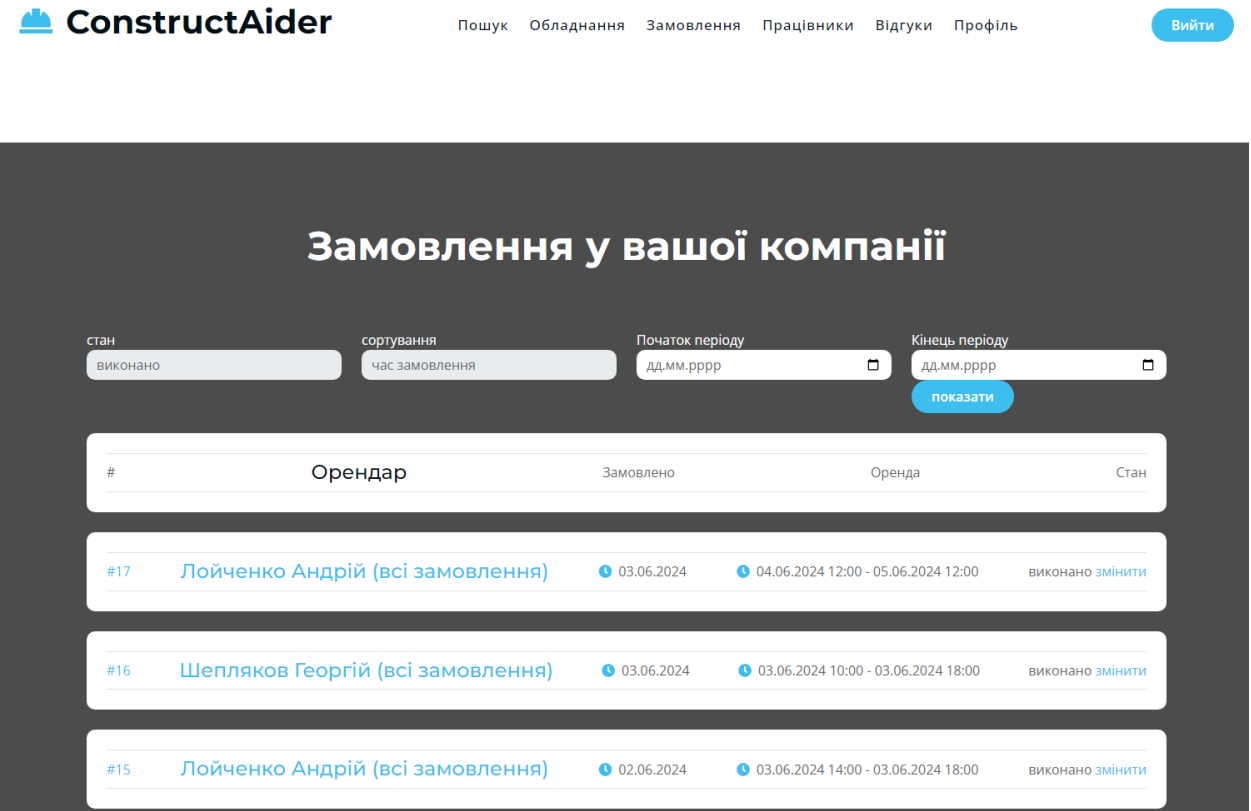


Рисунок 4.5 – Надання всіх замовлень компанії

За кожним замовленням можна натиснути на номер, тоді відобразиться коментар орендаря, якщо він є, та склад замовлення, його вартість. Якщо натиснути на ім'я орендаря, то відкриється окрема його сторінка. Якщо натиснути на запис зі всіма замовленнями поруч з ім'ям, то відкриється сторінка з замовленнями тільки цього орендаря у цього орендодавця. Якщо натиснути на кнопку змінити, то замість поточного стану відобразиться випадючий перелік, з якого можна обрати допустимий для такої зміни стан, підтвердивши його.

4.4 Висновки за розділом 4

Проведено тестування програми, перевірено в залежності від кожної

ролі користувача, чи працює кожна потрібна функція та яким чином. Тестування виявилось успішним, адже в підсумку всі результати відповідають тим, які очікувались.

ВИСНОВКИ

У роботі реалізовано розробку програмного забезпечення оренди обладнання для будівельних робіт, щоб забезпечити можливість взаємодії між різними орендодавцями та орендарями. Існуючі аналоги направлені на представлення конкретного орендодавця та оренду обладнання для будівельних робіт саме в нього, що звужує простір пошуку.

Розробку проведено як вебзастосунок з використанням Python-фреймворку Django, що спростило процес розробки за рахунок використання вбудованих рішень для забезпечення елементарних можливостей і концентрації на розробці найбільш особливих можливостей. Визначення конкретних можливостей, які програма для оренди обладнання для будівельних робіт має підтримувати, було здійснено для неавторизованого користувача, користувача-орендаря та користувача-працівника орендодавця. Структуру БД для збереження даних визначено і забезпечено використанням системи PostgreSQL.

У результаті розроблена програма, яка призначена для реалізації послуг з оренди обладнання для будівельних робіт, що підтримує потенційно можливості і короткострокової, і довгострокової оренди від різних орендодавців, забезпечує опрацювання не тільки загальних моделей обладнання для будівельних робіт у наявності в орендодавця, але і конкретних екземплярів, для яких можуть вноситися дані про виявлені вади, що сприяє прийняттю обґрунтованих рішень орендарем.

Виконана розробка дозволяє орендарям отримувати більшу кількість пропозицій за рахунок доступу до різних орендодавців одночасно, контролювати стан конкретних екземплярів обладнання, що сприяє зручності орендарів.

Усі завдання виконано, створено програму протестовано, що підтвердило успішну реалізацію.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Rental-Power – Інтернет Магазин Генераторів [Електрон. ресурс]. – Режим доступу : <https://rental-power.com.ua/ua/>.
2. United Rentals – Industrial & Construction Equipment Rentals & Tools [Electronic resource]. – Access mode : <https://www.unitedrentals.com/>.
3. BigRentz [Electronic resource] : The Nation's Largest Equipment Rental Company. – Access mode : <https://www.bigrentz.com/>.
4. Contributing to the Documentation (Symfony Docs) [Electronic resource]. – Access mode : <https://symfony.com/doc/current/contributing/documentation/overview.html>.
5. Документація Symfony [Електрон. ресурс]. – Режим доступу : <https://symfony.com.ua/doc/current/index.html>.
6. Django Tutorial in Visual Studio Code [Electronic resource]. – Access mode : <https://code.visualstudio.com/docs/python/tutorial-django>.
7. WaterLand – Water Park Website Template [Electronic resource]. – Access mode : <https://htmlcodex.com/water-park-website-template/>.

ДОДАТОК А
Технічне завдання

Вступ

Обладнання для будівельних робіт є достатньо часто дороговартісним, а тому не завжди воно є у власності тих команд, які зайняті будівництвом. Це зокрема стосується специфічного обладнання, яке використовується не завжди, а також випадків, коли команда невелика, але для виконання конкретного ремонту, будівництва потребує певного великогабаритного обладнання. Окрім того це стосується випадків, коли будівельні роботи виконуються самими власниками. Тоді з досить високою ймовірністю у власника щонайменше не все будівельне обладнання є в наявності. Тож у всіх цих випадках необхідно забезпечити можливість оренди обладнання для будівельних робіт, а джерелом такого обладнання можуть бути різні компанії-орендодавці. При створенні вебзастосунку для однієї конкретної такої компанії це значно звужує пошук, а створення вебзастосунку, який може бути використаний як певна платформа для оренди будівельного обладнання є перспективним практичним завданням.

А.1 Підстави для розробки

Дипломна кваліфікаційна робота виконувалась у відповідності та на підставі Наказу № 168 від 24 квітня 2024 р. за Національним університетом «Запорізька політехніка», яким визначено тему роботи – «Програмне забезпечення оренди обладнання для будівельних робіт».

А.2 Призначення розробки

Програма призначена для реалізації послуг з оренди обладнання для будівельних робіт, підтримуючи потенційно можливості і короткострокової, і довгострокової оренди від різних орендодавців, забезпечуючи опрацювання не тільки загальних моделей обладнання для будівельних робіт у наявності в

орендодавця, але і конкретних екземплярів, для яких можуть вноситися дані про виявлені вади, що сприяє прийняттю обґрунтованих рішень орендарем.

А.3 Основні вимоги до програми

А.3.1 Вимоги до функціональних характеристик

Функціональні характеристики програми оренди обладнання для будівельних робіт повинні включати:

- пошук обладнання для будівельних робіт для оренди;
- ознайомлення з моделлю обладнання для будівельних робіт;
- визначення вартості оренди обладнання для будівельних робіт;
- перегляд доступних екземплярів моделі обладнання для будівельних робіт;
- надання моделей обладнання для будівельних робіт у орендодавця;
- ознайомлення з орендодавцем обладнання для будівельних робіт;
- замовлення обладнання для будівельних робіт в оренду;
- авторизація і реєстрація користувачів;
- додавання моделі обладнання для будівельних робіт у орендодавця;
- редагування моделі обладнання для будівельних робіт у орендодавця;
- визначення виявленої вади екземпляра обладнання для будівельних робіт у орендодавця;
- підтримання або спростування виявленої вади екземпляра обладнання для будівельних робіт у орендодавця;
- додавання екземпляра моделі обладнання для будівельних робіт у орендодавця;
- редагування екземпляра моделі обладнання для будівельних робіт у орендодавця;
- виведення з оренди екземпляра моделі обладнання для будівельних робіт;
- ознайомлення з орендарем обладнання для будівельних робіт;

- редагування даних орендодавця обладнання для будівельних робіт;
- редагування даних орендаря обладнання для будівельних робіт;
- публікація відгуку про орендаря обладнання для будівельних робіт;
- публікація відгуку про модель обладнання для будівельних робіт у орендодавця;
- надання відгуків про задане модель обладнання для будівельних робіт у орендодавця;
- надання відгуків про орендаря обладнання для будівельних робіт;
- надання відгуків орендаря обладнання для будівельних робіт;
- надання відгуків орендодавця обладнання для будівельних робіт;
- надання працівників орендодавця обладнання для будівельних робіт;
- надання замовлень обладнання для будівельних робіт в оренду у орендодавця;
- надання власних замовлень обладнання для будівельних робіт в оренду;
- надання замовлень обладнання для будівельних робіт в оренду у орендодавця конкретним орендарем;
- керування замовленням обладнання для будівельних робіт в оренду;
- редагування базової вартості оренди моделі обладнання для будівельних робіт у орендодавця;
- додавання базової вартості оренди моделі обладнання для будівельних робіт у орендодавця;
- формування власного переліку обладнання для будівельних робіт;
- редагування власного переліку обладнання для будівельних робіт;
- додавання моделі обладнання для будівельних робіт у орендодавця до власного переліку;
- вилучення моделі обладнання для будівельних робіт у орендодавця з власного переліку;
- надання власного переліку обладнання для будівельних робіт.

А.3.2 Умови експлуатації

Обов'язковою умовою експлуатації програми оренди обладнання для будівельних робіт є наявність адміністратора програми або користувача, здатного виконувати такі функції. В обов'язки адміністратора входить наповнення даними. Також адміністратор повинен виконати розгортання додатку на сервері.

А.3.3 Вимоги до складу та параметрів технічних засобів

Технічне забезпечення сервера повинно включати мінімально:

- процесор Intel Xeon E5440 (частота 2,83 ГГц);
- жорсткий диск – 100 Гб (наявність фотографій може значно збільшувати вимоги до сервера, тому це потребує контролю та можливо введення обмежень на обсяг фотографій, які користувач може завантажувати, інакше необхідно розширювати потрібний простір);
- оперативна пам'ять – 4 Гб.

А.4 Порядок контролю та приймання

Починаючи з дати видачі завдання – 15 квітня, протягом наступних 8 тижнів має бути забезпечено виконання роботи за календарним планом, що містить в завданні на роботу. Календарний план визначає послідовність етапів виконання роботи та за кожним таким етапом – строк виконання і основний очікуваний результат. Після забезпечення цих очікуваних результатів формується пояснювальна записка до дипломної кваліфікаційної роботи. Остаточним кроком приймання роботи є її публічний захист.

ДОДАТОК Б
Текст програми

```

from .models import Constructequipment
from django.shortcuts import render
from .models import Constructequipmentphoto
from .models import Constructequipmentlessoruser
from .models import Constructequipmentrenter
from .models import Constructequipmentlistelement
from .models import Constructequipmentlist
from .models import Constructequipmentrentalelement
from .models import Constructequipmentrate
from .models import Constructequipmentproducer
from .models import Constructequipmentproperty
from .models import Constructequipmenttype
from .models import Constructequipmenttypeproperty
from .models import Constructequipmentsampleproblemapprove
from .models import Constructequipmentsampleproblem

def constructequipmentview(request, id_constructequipment,
constrental_start = None, constrental_unit = None,
constrental_duration = None):
    construct_dc = {}
    try:
        construct_equipment_full =
Constructequipment.objects.select_related("constructequipmentproducer")
        construct_equipment_full =
construct_equipment_full.select_related("constructequipmenttype")
        construct_equipment_full =
construct_equipment_full.select_related("constructequipmentlessor")
        construct_equipment_full =
construct_equipment_full.get(id = id_constructequipment)
    except Constructequipment.DoesNotExist:
        return render(request, "constructaider/error.html",
{"prompt": "Вибачте, але ви задали неіснуючу модель обладнання для
будівельних робіт. Дивно, що це сталося, але це могло відбутися з двох
причин. Можливо, ви самостійно змінили адресу. Будьте уважні. Можливо
ви перейшли за цим посиланням на самій сторінці. Тоді просимо вас
надати цю інформацію адміністратору для виправлення помилки. Дякуємо
за розуміння і терпіння. Бажаємо успішної роботи з нами!"})
        construct_dc["equipment"] = construct_equipment_full
        construct_equipment_photos =
Constructequipmentphoto.objects.filter(constructequipment =
construct_equipment_full)
        construct_dc["photos"] = construct_equipment_photos

```

```

        construct_equipment_properties =
Constructequipproperty.objects.select_related("constructequiptypeprope
rty")
        construct_equipment_properties =
construct_equipment_properties.select_related("constructequiptypeprope
rty__constructequiptype")
        construct_equipment_properties =
construct_equipment_properties.filter(constructequiptypeproperty__cons
tructequipment = construct_equipment_full)
        construct_dc["properties"] = construct_equipment_properties
        construct_equipment_samples =
Constructequipmentsample.objects.filter(constructequipment =
construct_equipment_full)
        for construct_equipment_sample in
construct_equipment_samples:
            sampleproblems =
Constructequipmentsampleproblem.objects.select_related("constructequip
renter")
            sampleproblems =
sampleproblems.filter(constructequipmentsample =
construct_equipment_sample)
            sampleproblems = sampleproblems.order_by("-
problem_date")
            for sampleproblem in sampleproblems:
                approves =
Constructequipmentsampleproblemapprove.objects.select_related("constru
ctequiprenter")
                approves =
approves.filter(constructequipmentsampleproblem = sampleproblem)
                approves = approves.order_by("-id")
                sampleproblem.approves = approves
                construct_equipment_sample.problems = sampleproblems
                lessor_user = None
                if not request.user.is_active:
                    construct_equipment_samples =
construct_equipment_samples.filter(active_rent = True)
                else:
                    lessor_user =
Constructequiplessoruser.objects.filter(account = request.user)
                    if not lessor_user:
                        construct_equipment_samples =
construct_equipment_samples.filter(active_rent = True)

```

```

        renter_users =
Constructequiprenter.objects.filter(account = request.user)
        if renter_users:
            renter_user = renter_users[0]
            renter_lists =
Constructequiplist.objects.filter(constructequiprenter = renter_user)
            for renter_list in renter_lists:
                equipment_in_list =
Constructequiplistelement.objects.filter(constructequiplist =
renter_list, constructequipment = construct_equipment_full)
                if equipment_in_list:
                    renter_list.constrequeiment = True
                else:
                    renter_list.constrequeiment = False
            construct_dc["lists"] = renter_lists
            construct_dc["samples"] = construct_equipment_samples
            import calendar
            import datetime
            if not lessor_user:
                if not constrental_start:
                    constrental_start = datetime.datetime.now() +
datetime.timedelta(days=1)
                    constrental_unit = 0
                    constrental_duration = 1
                    constrental_end = constrental_start +
datetime.timedelta(hours=1)
                else:
                    if constrental_unit == 0:
                        constrental_end = constrental_start +
datetime.timedelta(hours=constrental_duration)
                    elif constrental_unit == 1:
                        constrental_end = constrental_start +
datetime.timedelta(days=constrental_duration)
                    elif constrental_unit == 2:
                        constrental_end = constrental_start +
datetime.timedelta(weeks=constrental_duration)
                    elif constrental_unit == 3:
                        if constrental_start.month != 12:
                            month_days =
calendar.monthrange(constrental_start.year, constrental_start.month)
                            if month_days < constrental_start.day:

```

```

        constrental_end =
datetime.datetime(constrental_start.year, constrental_start.month,
month_days)
        else:
            constrental_end =
datetime.datetime(constrental_start.year, constrental_start.month,
constrental_start.day)
        else:
            constrental_end =
datetime.datetime(constrental_start.year+1, 1, constrental_start.day)
            possible_construct_equipment_samples = []
            for construct_equipment_sample in
construct_equipment_samples:
                construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__lte =
constrental_start, constructequiprental__rental__rental_end__gte =
constrental_start)
                if not construct_equipment_sample_in_orders:
                    construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__gte =
constrental_start, constructequiprental__rental__rental_end__lte =
constrental_end)
                if not construct_equipment_sample_in_orders:
                    construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__gte =
constrental_start, constructequiprental__rental__rental_end__gte =
constrental_end)
            else:

possible_construct_equipment_samples.append(construct_equipment_sample
)

        constrental_date = datetime.datetime.now()
        for possible_construct_equipment in
possible_construct_equipment_samples:
            equipment_rates =
Constructequirate.objects.filter(constructequipment =

```

```

construct_equipment_sample.constructequipment, active_date__lte =
constrental_date)
        equipment_rates =
equipment_rates.order_by("active_date", "min_rate")
        equipment_rates = equipment_rates.values("min_rate",
"price_rate", "price_unit", "active_date")
        last_date = equipment_rates[-1]
        last_date = last_date.active_date
        if constrental_unit == 0:
            constrental_hours_duration =
constrental_duration
        elif constrental_unit == 1:
            constrental_hours_duration =
constrental_duration * 24
        elif constrental_unit == 2:
            constrental_hours_duration =
constrental_duration * 24 * 7
        elif constrental_unit == 3:
            constrental_hours = constrental_end -
constrental_start
            constrental_hours_duration =
constrental_hours.weeks * 24 * 7 + constrental_hours.days * 24 +
constrental_hours.hours
            last_rate = None
            rate_hours_duration_saved = 0
            for equipment_rate in equipment_rates[::-1]:
                if equipment_rate.active_date == last_date:
                    if equipment_rate.price_unit == 0:
                        rate_hours_duration =
equipment_rate.min_rate
                    elif equipment_rate.price_unit == 1:
                        rate_hours_duration =
equipment_rate.min_rate * 24
                    elif equipment_rate.price_unit == 2:
                        rate_hours_duration =
equipment_rate.min_rate * 24 * 7
                    elif equipment_rate.price_unit == 3:
                        rate_hours_duration =
equipment_rate.min_rate * 30 * 24
                    if rate_hours_duration <=
constrental_hours_duration and rate_hours_duration >
rate_hours_duration_saved:

```

```

rate_hours_duration_saved =
rate_hours_duration
last_rate = equipment_rate
else:
break
if last_rate.price_unit == 0:
possible_constrental_price = constrental_end -
constrental_start
possible_constrental_price =
possible_constrental_price.hours
possible_constrental_price =
possible_constrental_price * last_rate.price_rate
elif last_rate.price_unit == 1:
possible_constrental_price = constrental_end -
constrental_start
possible_constrental_price =
possible_constrental_price.days
possible_constrental_price =
possible_constrental_price * last_rate.price_rate
elif last_rate.price_unit == 2:
possible_constrental_price = constrental_end -
constrental_start
possible_constrental_price =
possible_constrental_price.weeks
possible_constrental_price =
possible_constrental_price * last_rate.price_rate
elif last_rate.price_unit == 3:
if constrental_end.year ==
constrental_start.year and constrental_end.day >=
constrental_start.day:
possible_constrental_price =
constrental_end.month - constrental_start.month + 1
elif constrental_end.year ==
constrental_start.year:
possible_constrental_price =
constrental_end.month - constrental_start.month
elif constrental_end.day >=
constrental_start.day:
possible_constrental_price =
constrental_end.month + 12 - constrental_start.month + 12 *
(constrental_end.year - constrental_start.year - 1) + 1
else:

```

```

        possible_constrental_price =
constrental_end.month + 12 - constrental_start.month + 12 *
(constrental_end.year - constrental_start.year -1) + 1
        possible_constrental_price =
possible_constrental_price * last_rate.price_rate
        construct_dc["price"] = possible_constrental_price
        construct_dc["possible_samples"] =
possible_construct_equipment_samples
        return render(request,
"constructaider/constructequipment.html", construct_dc)

def searchconstructequipmentview(request):
    construct_dc = {}
    construct equipments_full =
Constructequipment.objects.select_related("constructequipproducer")
    construct equipments_full =
construct equipments_full.select_related("constructequiptype")
    construct equipments_full =
construct equipments_full.select_related("constructequiplessor")
    construct equipments_full = construct equipments_full.all()
    constructproducer = self.request.GET.get("producer", None)
    construct_dc["producers"] =
Constructequipproducer.objects.all()
    if constructproducer:
        constructproducer = int(constructproducer)
        construct equipments_full =
construct equipments_full.filter(constructequipproducer__id =
constructproducer)
        constructequipmenttype = self.request.GET.get("equiptype",
None)
        if constructequipmenttype:
            constructequipmenttype = int(constructequipmenttype)
            construct equipments_full =
construct equipments_full.filter(constructequiptype__id =
constructequipmenttype)
            constructequipmentmodel = self.request.GET.get("equipmodel",
None)
            construct_dc["types"] = Constructequiptype.objects.all()
            properties =
Constructequiptypeproperty.objects.select_related("constructequiptype"
)
            properties = properties.all()
            for property_constr in properties:

```

```

        property_constr.values =
Constructequipproperty.objects.filter(constructequiptypeproperty =
property_constr).distinct("propertyvalue")
        construct_dc["properties"] = properties
        if constructequipmentmodel:
            construct equipments_full =
construct equipments_full.filter(equipment_model__icontains =
constructequipmentmodel)
            minaverage_opinion = self.request.GET.get("opinion", None)
            if minaverage_opinion:
                minaverage_opinion = float(minaverage_opinion)
                construct equipments_full =
construct equipments_full.filter(average_opinion__gte =
minaverage_opinion)
            import calendar
            import datetime
            constrental_start = self.request.GET.get("renstart", None)
            constrental_duration = self.request.GET.get("duration",
None)
            constrental_unit = self.request.GET.get("unit", None)
            if not constrental_start:
                constrental_start = datetime.datetime.now() +
datetime.timedelta(days=1)
                constrental_unit = 0
                constrental_duration = 1
                constrental_end = constrental_start +
datetime.timedelta(hours=1)
            else:
                constrental_start = datetime.strptime(constrental_start,
'%d.%m.%y %H:%M')
                if constrental_unit:
                    constrental_unit = int(constrental_unit)
                else:
                    constrental_unit = 0
                if constrental_duration:
                    constrental_duration = int(constrental_duration)
                else:
                    constrental_duration = 1
                if constrental_unit == 0:
                    constrental_end = constrental_start +
datetime.timedelta(hours=constrental_duration)
                elif constrental_unit == 1:

```

```

        constrental_end = constrental_start +
datetime.timedelta(days=constrental_duration)
        elif constrental_unit == 2:
            constrental_end = constrental_start +
datetime.timedelta(weeks=constrental_duration)
        elif constrental_unit == 3:
            if constrental_start.month != 12:
                month_days =
calendar.monthrange(constrental_start.year, constrental_start.month)
                if month_days < constrental_start.day:
                    constrental_end =
datetime.datetime(constrental_start.year, constrental_start.month,
month_days)
            else:
                constrental_end =
datetime.datetime(constrental_start.year, constrental_start.month,
constrental_start.day)
        else:
            constrental_end =
datetime.datetime(constrental_start.year+1, 1, constrental_start.day)
            constrental_city = self.request.GET.get("city", None)
            if constrental_city:
                construct equipments_full =
construct equipments_full.filter(constructequiplessor__function_city =
constrental_city)
                constrental_properties = self.request.GET.get("properties",
None)
                constrental_properties = constrental_properties.split("--")
                proc_constrental_properties = []
                for constrental_property in constrental_properties:
                    constrental_property_namevalue =
constrental_property.split("-")
                    constrental_property_name =
constrental_property_namevalue[0]
                    constrental_property_value =
constrental_property_namevalue[1]

                proc_constrental_properties.append((constrental_property_name,
constrental_property_value))
                construct equipments_full =
construct equipments_full.order_by("-average_opinion")
                constrental_minprice = self.request.GET.get("minprice",
None)

```

```

        construct equipments_full_temp = []
        for construct equipment_full in construct equipments_full:
            all_properties = True
            for proc_constrental_property in
proc_constrental_properties:
                ps =
Constructequipproperty.objects.filter(constructequiptypeproperty__id =
int(proc_constrental_property[0]), propertyvalue =
proc_constrental_property[1])
                if not ps:
                    all_properties = False
                    break

construct equipments_full_temp.append(construct equipment_full)
construct equipments_full = construct equipments_full_temp
if constrental_minprice:
    constrental_minprice = float(constrental_minprice)
else:
    constrental_minprice = 0
constrental_maxprice = self.request.GET.get("maxprice",
None)
if constrental_maxprice:
    constrental_maxprice = float(constrental_minprice)
else:
    constrental_maxprice = 1000000000
possible_construct equipment = []
for construct equipment_full in construct equipments_full:
    construct equipment_samples =
Constructequipmentsample.objects.filter(constructequipment =
construct equipment_full, active_rent = True)
    for construct equipment_sample in
construct equipment_samples:
        construct equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__lte =
constrental_start, constructequiprental__rental__rental_end__gte =
constrental_start)
        if not construct equipment_sample_in_orders:
            construct equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__gte =

```

```

constrental_start, constructequiprental__rental__rental_end__lte =
constrental_end)
        if not construct_equipment_sample_in_orders:
            construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__gte =
constrental_start, constructequiprental__rental__rental_end__gte =
constrental_end)
        else:

possible_construct_equipment_samples.append(construct_equipment_sample
)

            break
            constrental_date = datetime.datetime.now()
            for construct_equipment_sample in
possible_construct_equipment_samples[0]:
                equipment_rates =
Constructequirate.objects.filter(constructequipment =
construct_equipment_sample.constructequipment, active_date__lte =
constrental_date)
                equipment_rates =
equipment_rates.order_by("active_date", "min_rate")
                equipment_rates = equipment_rates.values("min_rate",
"price_rate", "price_unit", "active_date")
                last_date = equipment_rates[-1]
                last_date = last_date.active_date
                if constrental_unit == 0:
                    constrental_hours_duration =
constrental_duration
                elif constrental_unit == 1:
                    constrental_hours_duration =
constrental_duration * 24
                elif constrental_unit == 2:
                    constrental_hours_duration =
constrental_duration * 24 * 7
                elif constrental_unit == 3:
                    constrental_hours = constrental_end -
constrental_start
                    constrental_hours_duration =
constrental_hours.weeks * 24 * 7 + constrental_hours.days * 24 +
constrental_hours.hours
                    last_rate = None

```

```

        rate_hours_duration_saved = 0
        for equipment_rate in equipment_rates[::-1]:
            if equipment_rate.active_date == last_date:
                if equipment_rate.price_unit == 0:
                    rate_hours_duration =
equipment_rate.min_rate
                    elif equipment_rate.price_unit == 1:
                        rate_hours_duration =
equipment_rate.min_rate * 24
                    elif equipment_rate.price_unit == 2:
                        rate_hours_duration =
equipment_rate.min_rate * 24 * 7
                    elif equipment_rate.price_unit == 3:
                        rate_hours_duration =
equipment_rate.min_rate * 30 * 24
                        if rate_hours_duration <=
constrental_hours_duration and rate_hours_duration >
rate_hours_duration_saved:
                            rate_hours_duration_saved =
rate_hours_duration
                            last_rate = equipment_rate
                        else:
                            break
                    if last_rate.price_unit == 0:
                        possible_constrental_price = constrental_end -
constrental_start
                            possible_constrental_price =
possible_constrental_price.hours
                            possible_constrental_price =
possible_constrental_price * last_rate.price_rate
                        elif last_rate.price_unit == 1:
                            possible_constrental_price = constrental_end -
constrental_start
                            possible_constrental_price =
possible_constrental_price.days
                            possible_constrental_price =
possible_constrental_price * last_rate.price_rate
                        elif last_rate.price_unit == 2:
                            possible_constrental_price = constrental_end -
constrental_start
                            possible_constrental_price =
possible_constrental_price.weeks

```

```

        possible_constrental_price =
possible_constrental_price * last_rate.price_rate
        elif last_rate.price_unit == 3:
            if constrental_end.year ==
constrental_start.year and constrental_end.day >=
constrental_start.day:
                possible_constrental_price =
constrental_end.month - constrental_start.month + 1
                elif constrental_end.year ==
constrental_start.year:
                    possible_constrental_price =
constrental_end.month - constrental_start.month
                    elif constrental_end.day >=
constrental_start.day:
                        possible_constrental_price =
constrental_end.month + 12 - constrental_start.month + 12 *
(constrental_end.year - constrental_start.year -1) + 1
                        else:
                            possible_constrental_price =
constrental_end.month + 12 - constrental_start.month + 12 *
(constrental_end.year - constrental_start.year -1) + 1
                            possible_constrental_price =
possible_constrental_price * last_rate.price_rate
                            construct_equipment_full.price =
possible_constrental_price
                            if possible_constrental_price >=
constrental_minprice and possible_constrental_price <=
constrental_maxprice:

possible_construct_equipment.append(construct_equipment_full)
        construct_dc["possible_equipment"] =
possible_construct_equipment
        renter_lists = []
        if request.user.is_active:
            renter_users =
Constructequiprenter.objects.filter(account = request.user)
            if renter_users:
                renter_user = renter_users[0]
                renter_lists =
Constructequiplist.objects.filter(constructequiprenter = renter_user)
                for renter_list in renter_lists:
                    renter_list.equipments = []

```

```

        for possible_construct_equip in
possible_construct_equipment:
            equipment_in_list =
Constructequiplistelement.objects.filter(constructequiplist =
renter_list, constructequipment = possible_construct_equip)
            if equipment_in_list:
                renter_list.equipments.append(True)
            else:
                renter_list.equipments.append(False)
        construct_dc["lists"] = renter_lists
        return render(request,
"constructaider/searchconstructequipment.html", construct_dc)

def constructequipmentrentalgetview(request):
    construct_dc = {}
    constrental_equipment_type = self.request.GET.get("type",
None)
    if constrental_equipment_type:
        constrental_equipment_type =
int(constrental_equipment_type)
    else:
        constrental_equipment_type = 0
    constrental_equipments = self.request.GET.get("equipment",
None)
    constrental_equipments = constrental_equipments.split("-")
    constrental_ready_equipments = []
    for constrental_equipment in constrental_equipments:
        if not constrental_equipment_type:
            try:
                construct_equipment_full =
Constructequipment.objects.select_related("constructequiproducer")
                construct_equipment_full =
construct_equipment_full.select_related("constructequiptype")
                construct_equipment_full =
construct_equipment_full.select_related("constructequiplessor")
                construct_equipment_full =
construct_equipment_full.get(id = int(constrental_equipment))
            except Constructequipment.DoesNotExist:
                return render(request,
"constructaider/error.html", {"prompt": "Вибачте, але ви задали
неіснуючу модель обладнання для будівельних робіт. Дивно, що це
сталось, але це могло відбутися з двох причин. Можливо, ви самостійно
змінити адресу. Будьте уважні. Можливо ви перейшли за цим посиланням

```

на самій сторінці. Тоді просимо вас надати цю інформацію адміністратору для виправлення помилки. Дякуємо за розуміння і терпіння. Бажаємо успішної роботи з нами!"})

```

constrental_ready equipments.append(construct_equipment_full)
    else:
        try:
            construct_equipment_full =
Constructequipmentsample.objects.select_related("constructequipment__c
onstructequipprouducer", active_rent = True)
            construct_equipment_full =
construct_equipment_full.select_related("constructequipment__construct
equiptype")
            construct_equipment_full =
construct_equipment_full.select_related("constructequipment__construct
equiplessor")
            construct_equipment_full =
construct_equipment_full.get(id = int(constrental_equipment))
        except Constructequipmentsample.DoesNotExist:
            return render(request,
"constructaider/error.html", {"prompt": "Вибачте, але ви задали
неіснуючий екземпляр обладнання для будівельних робіт. Дивно, що це
сталось, але це могло відбутися з двох причин. Можливо, ви самостійно
змінити адресу. Будьте уважні. Можливо ви перейшли за цим посиланням
на самій сторінці. Тоді просимо вас надати цю інформацію
адміністратору для виправлення помилки. Дякуємо за розуміння і
терпіння. Бажаємо успішної роботи з нами!"})

constrental_ready equipments.append(construct_equipment_full)
    if not constrental_equipment_type:
        construct_dc["equipment"] = constrental_ready equipments
    import calendar
    import datetime
    constrental_start = self.request.GET.get("renstart", None)
    constrental_duration = self.request.GET.get("duration",
None)
    constrental_unit = self.request.GET.get("unit", None)
    if not constrental_start:
        constrental_start = datetime.datetime.now() +
datetime.timedelta(days=1)
        constrental_unit = 0
        constrental_duration = 1

```

```

        constrental_end = constrental_start +
datetime.timedelta(hours=1)
    else:
        constrental_start = datetime.strptime(constrental_start,
'%d.%m.%y %H:%M')
        if constrental_unit:
            constrental_unit = int(constrental_unit)
        else:
            constrental_unit = 0
        if constrental_duration:
            constrental_duration = int(constrental_duration)
        else:
            constrental_duration = 1
        if constrental_unit == 0:
            constrental_end = constrental_start +
datetime.timedelta(hours=constrental_duration)
        elif constrental_unit == 1:
            constrental_end = constrental_start +
datetime.timedelta(days=constrental_duration)
        elif constrental_unit == 2:
            constrental_end = constrental_start +
datetime.timedelta(weeks=constrental_duration)
        elif constrental_unit == 3:
            if constrental_start.month != 12:
                month_days =
calendar.monthrange(constrental_start.year, constrental_start.month)
                if month_days < constrental_start.day:
                    constrental_end =
datetime.datetime(constrental_start.year, constrental_start.month,
month_days)
            else:
                constrental_end =
datetime.datetime(constrental_start.year, constrental_start.month,
constrental_start.day)
        else:
            constrental_end =
datetime.datetime(constrental_start.year+1, 1, constrental_start.day)
        if constrental_equipment_type:
            constrental_ready equipments_samples =
constrental_ready equipments
        else:
            constrental_ready equipments_samples = []

```

```

        for construct_equipment_full in
constrental_ready equipments:
            construct_equipment_samples =
Constructequipmentsample.objects.filter(constructequipment =
construct_equipment_full, active_rent = True)
            for construct_equipment_sample in
construct_equipment_samples:
                construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__lte =
constrental_start, constructequiprental__rental__rental_end__gte =
constrental_start)
                if not construct_equipment_sample_in_orders:
                    construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__gte =
constrental_start, constructequiprental__rental__rental_end__lte =
constrental_end)
                if not construct_equipment_sample_in_orders:
                    construct_equipment_sample_in_orders =
Constructequiprentalelement.objects.filter(constructequipmentsample =
construct_equipment_sample, constructequiprental__rental_state__in =
[1, 3, 4, 5], constructequiprental__rental_start__gte =
constrental_start, constructequiprental__rental__rental_end__gte =
constrental_end)
                else:

constrental_ready equipments_samples.append(construct_equipment_sample
)

                break
            constrental_date = datetime.datetime.now()
            total_rental_price = 0
            for constrental_ready equipments_sample in
constrental_ready equipments_samples:
                equipment_rates =
Constructequirate.objects.filter(constructequipment =
constrental_ready equipments_sample.constructequipment,
active_date__lte = constrental_date)
                equipment_rates =
equipment_rates.order_by("active_date", "min_rate")

```

```

        equipment_rates = equipment_rates.values("min_rate",
"price_rate", "price_unit", "active_date")
        last_date = equipment_rates[-1]
        last_date = last_date.active_date
        if constrental_unit == 0:
            constrental_hours_duration = constrental_duration
        elif constrental_unit == 1:
            constrental_hours_duration = constrental_duration *
24
        elif constrental_unit == 2:
            constrental_hours_duration = constrental_duration *
24 * 7
        elif constrental_unit == 3:
            constrental_hours = constrental_end -
constrental_start
            constrental_hours_duration = constrental_hours.weeks
* 24 * 7 + constrental_hours.days * 24 + constrental_hours.hours
            last_rate = None
            rate_hours_duration_saved = 0
            for equipment_rate in equipment_rates[::-1]:
                if equipment_rate.active_date == last_date:
                    if equipment_rate.price_unit == 0:
                        rate_hours_duration =
equipment_rate.min_rate
                    elif equipment_rate.price_unit == 1:
                        rate_hours_duration =
equipment_rate.min_rate * 24
                    elif equipment_rate.price_unit == 2:
                        rate_hours_duration =
equipment_rate.min_rate * 24 * 7
                    elif equipment_rate.price_unit == 3:
                        rate_hours_duration =
equipment_rate.min_rate * 30 * 24
                    if rate_hours_duration <=
constrental_hours_duration and rate_hours_duration >
rate_hours_duration_saved:
                        rate_hours_duration_saved =
rate_hours_duration
                    last_rate = equipment_rate
                else:
                    break
            if last_rate.price_unit == 0:

```

```

        possible_constrental_price = constrental_end -
constrental_start
        possible_constrental_price =
possible_constrental_price.hours
        possible_constrental_price =
possible_constrental_price * last_rate.price_rate
        elif last_rate.price_unit == 1:
            possible_constrental_price = constrental_end -
constrental_start
            possible_constrental_price =
possible_constrental_price.days
            possible_constrental_price =
possible_constrental_price * last_rate.price_rate
            elif last_rate.price_unit == 2:
                possible_constrental_price = constrental_end -
constrental_start
                possible_constrental_price =
possible_constrental_price.weeks
                possible_constrental_price =
possible_constrental_price * last_rate.price_rate
            elif last_rate.price_unit == 3:
                if constrental_end.year == constrental_start.year
and constrental_end.day >= constrental_start.day:
                    possible_constrental_price =
constrental_end.month - constrental_start.month + 1
                    elif constrental_end.year == constrental_start.year:
                        possible_constrental_price =
constrental_end.month - constrental_start.month
                        elif constrental_end.day >= constrental_start.day:
                            possible_constrental_price =
constrental_end.month + 12 - constrental_start.month + 12 *
(constrental_end.year - constrental_start.year -1) + 1
                        else:
                            possible_constrental_price =
constrental_end.month + 12 - constrental_start.month + 12 *
(constrental_end.year - constrental_start.year -1) + 1
                            possible_constrental_price =
possible_constrental_price * last_rate.price_rate
                            constrental_ready equipments_sample.price =
possible_constrental_price
                            total_rental_price = total_rental_price +
possible_constrental_price
                            construct_dc["price"] = total_rental_price

```

```
        construct_dc["samples"] =  
constrental_ready equipments_samples  
        return render(request,  
"constructaider/grentalconstructequipment.html", construct_dc)
```

ДОДАТОК В
Слайди презентації



Рисунок В.1 – Перший слайд



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

Порівняння розробки програми оренди обладнання для будівельних робіт

Критерій	Rental-Power	United Rentals	Моя програма
Пошук обладнання для будівельних робіт	За назвою	За назвою, категорією та з визначенням міста, періоду оренди	За назвою, категорією, виробником, властивостями, з визначенням міста, періоду оренди, обмеженням оцінки, вартості
Джерело оренди	Компанія Rental-Power	Компанія United Rentals	Різні зареєстровані компанії-орендодавці
Період оренди	Півдоби або кількість днів	Кількість днів, тижнів, місяців	Кількість годин, днів, тижнів, місяців
Відгуки	Про обладнання	Відсутні	Про обладнання, про орендарів
Визначення вад екземплярів обладнання	Відсутнє	Відсутнє	Наявне

3

Рисунок В.3 – Порівняння розробки програми оренди обладнання для будівельних робіт

Вибір фреймворку для розробки програми оренди обладнання для будівельних робіт

Критерій	Symfony	Django
Підхід до побудови	Модульний	Монолітний
Мова	PHP	Python
Шаблон проєктування	Model-View-Controller, але з наявністю шаблонів	MVT
Рушій підтримки шаблонів	Twig	Django
Наявність підтримки	Менш розвинена	Більш розвинена
Засоби забезпечення безпеки	Аутентифікація, збереження паролів, контроль доступу за ролями	Аутентифікація, збереження паролів, контроль доступу за ролями, SQL-ін'єкції та багато інших

4

Рисунок В.4 – Вибір фреймворку для розробки програми оренди обладнання для будівельних робіт

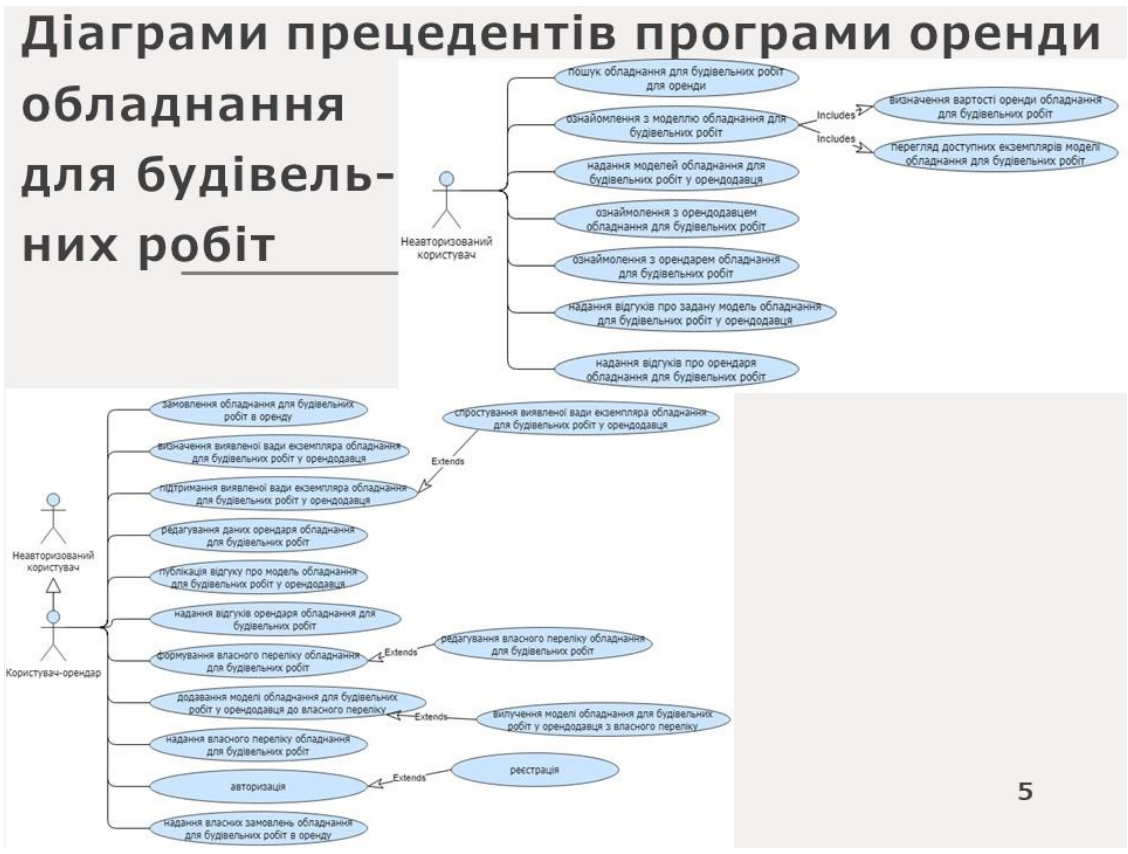


Рисунок В.5 – Діаграми прецедентів програми оренди обладнання для будівельних робіт

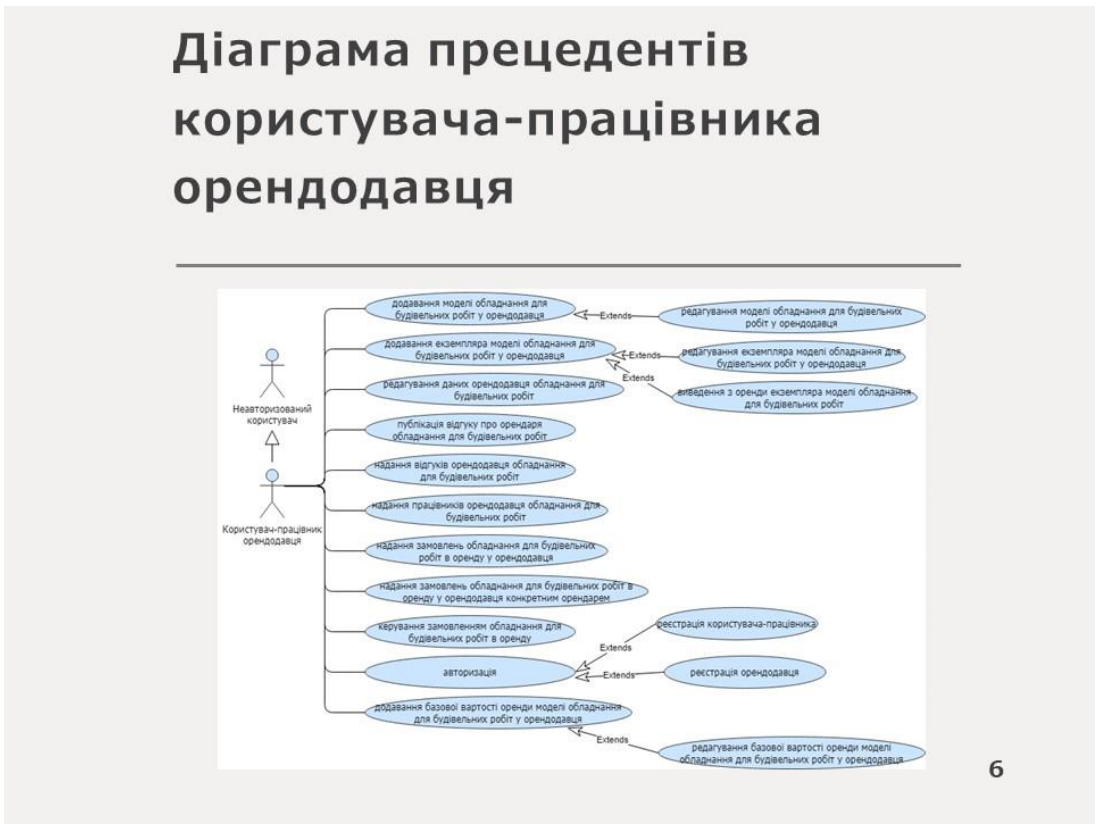
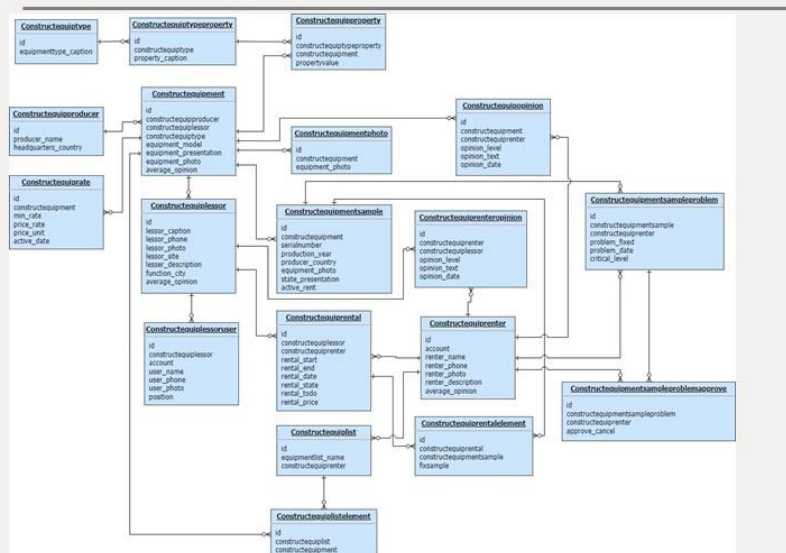


Рисунок В.6 – Діаграма прецедентів користувача-працівника орендодавця

Схема бази даних програми оренди обладнання для будівельних робіт



7

Рисунок В.7 – Схема бази даних програми оренди обладнання для будівельних робіт

Схема пошуку обладнання для будівельних робіт для оренди



8

Рисунок В.8 – Схема пошуку обладнання для будівельних робіт для оренди

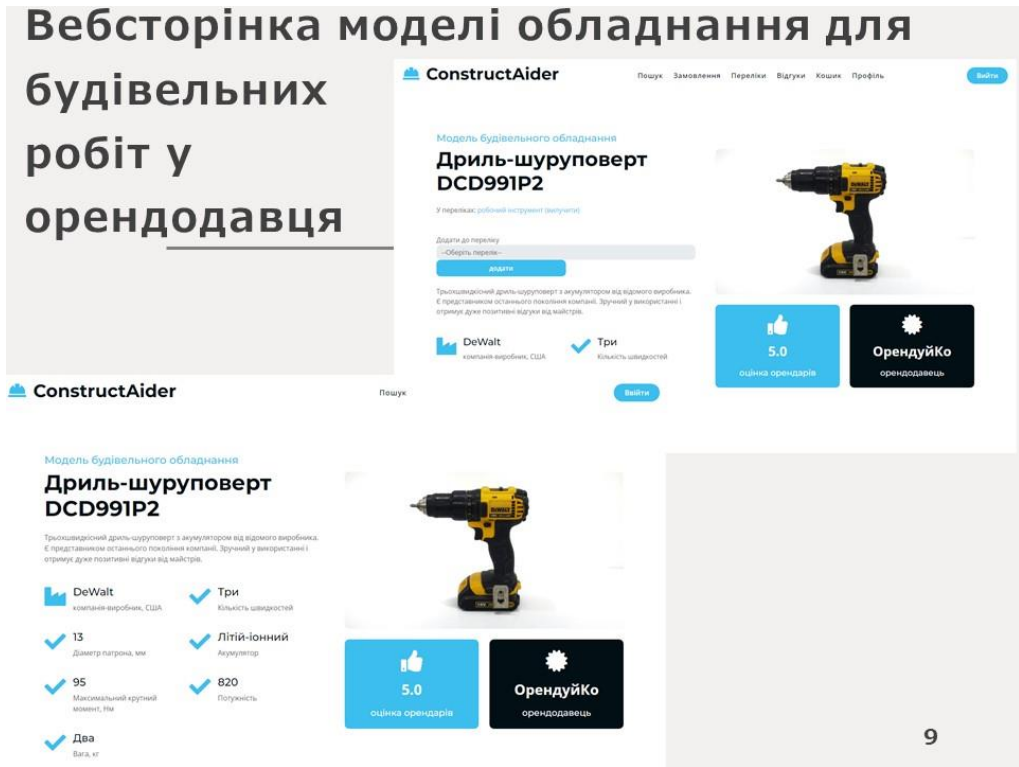


Рисунок В.9 – Вебсторінка моделі обладнання для будівельних робіт у орендодавця

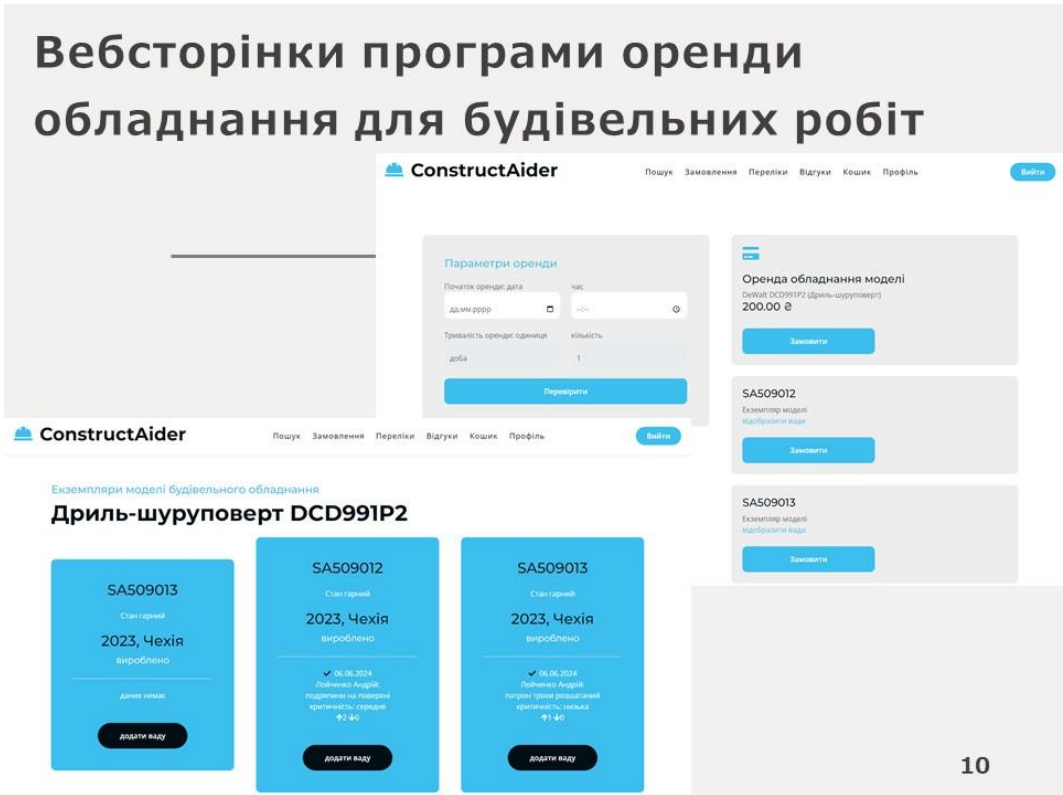
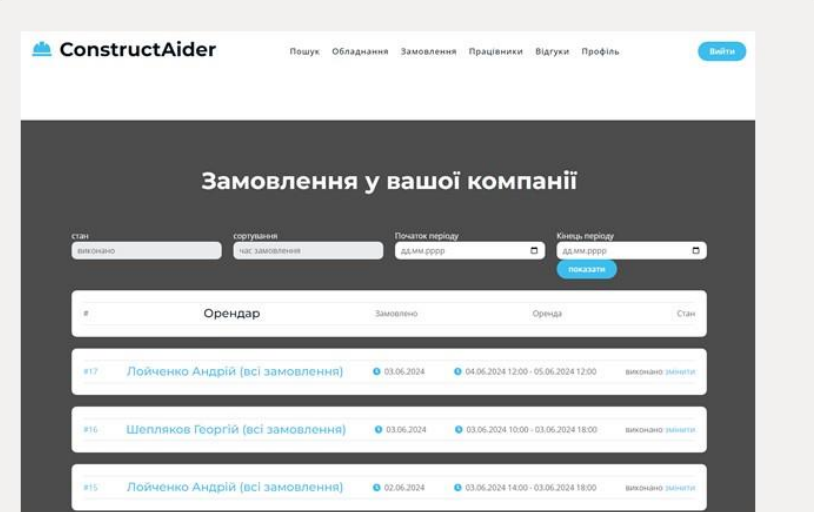


Рисунок В.10 – Вебсторінки програми оренди обладнання для будівельних робіт

Вебсторінка надання всіх замовлень компанії-орендодавця



11

Рисунок В.11 – Вебсторінка надання всіх замовлень компанії-орендодавця

Висновки

У роботі реалізовано розробку програмного забезпечення оренди обладнання для будівельних робіт, щоб забезпечити можливість взаємодії між різними орендодавцями та орендарями. Існуючі аналоги напарвлені на представлення конкретного орендодавця та оренду обладнання для будівельних робіт саме в нього, що звужує простір пошуку.

Розробку проведено як вебзастосунок з використанням Python-фреймворку Django, що спростило процес розробки за рахунок використання вбудованих рішень для забезпечення елементарних можливостей і концентрації на розробці найбільш особливих можливостей. Визначення конкретних можливостей, які програма для оренди обладнання для будівельних робіт має підтримувати, було здійснено для неавторизованого користувача, користувача-орендаря та користувача-працівника орендодавця. Структуру бази даних для збереження даних визначено і забезпечено використанням системи PostgreSQL.

Виконана розробка дозволяє орендарям отримувати більшу кількість пропозицій за рахунок доступу до різних орендодавців одночасно, контролювати стан конкретних екземплярів обладнання, що сприяє зручності орендарів.

Усі завдання виконано, створено програму протестовано, що підтвердило успішну реалізацію.

12

Рисунок В.12 – Висновки