МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

факультет програмної інженерії та бізнесу

кафедра інженерії програмного забезпечення

**Курсова робота**

з дисципліни «Об’єкто-орієнтоване програмування»

*назва дисципліни*

на тему:«Розроблення програмного забезпечення з використанням

об’єктно-орієнтованої парадигми»

Виконав: студентка 1 курсу групи № 612Пст

напряму підготовки (спеціальності)

121 інженерія програмного забезпечення

(шифр і назва напряму підготовки (спеціальності))

Кіян Л.В.

(прізвище й ініціали студента)

Керівник:

(прізвище й ініціали)

Національна шкала:

Кількість балів:

Оцінка ECTS:

Члени комісії

(підпис) (прізвище й ініціали)

(підпис) (прізвище й ініціали)

Харків – 2023

# Зміст

# Вступ

У сучасному світі, де швидкість і ефективність є ключовими факторами, логістичні послуги стають невід’ємною частиною нашого повсякденного життя. Відправлення пошти, доставка товарів, логістика виробництва - це всі сфери, де важливо забезпечити оптимальний рівень обслуговування та мінімізувати часові затрати.

Програма Express Post вирішує кілька важливих проблем. По-перше, це ефективність доставки. Однією з ключових проблем є оптимізація маршрутів та вибір найшвидшого способу доставки. Ця програма допоможе вирішити цю проблему, обчислюючи оптимальний маршрут для кожного відправлення. Друга проблема - відстеження відправлень. Клієнти хочуть бачити, де знаходиться їх посилка в реальному часі. Express Post надає можливість відстежувати статус відправлення, що підвищує довіру клієнтів та забезпечує їх спокій. Третя проблема - безпека та конфіденційність. Забезпечення безпеки відправлень та конфіденційності даних - ще одна важлива проблема. Express Post використовує шифрування та захист даних, щоб гарантувати безпеку користувачів.

Важливість програми Express Post важко переоцінити. Швидкість - це ключовий аспект в сучасному світі. Люди цінують свій час і очікують, що послуги будуть надаватися оперативно та ефективно. Express Post, зі своїм слоганом “Пошта, яка цінує ваш час”, відповідає цим вимогам, надаючи швидку та надійну послугу доставки.

Дизайн програми також важливий. Мінімалістичний дизайн Express Post відображає його функціональність та простоту використання.

ExpressPost - це не просто програма пошти, це інструмент, який допоможе ефективно вирішувати логістичні завдання та зберегти ваш час.

# 1. Розроблення функціональних вимог до програмного забезпечення

## Аналіз аналогічних програмних систем

### 1. Інтерфейс та замовлення:

− Delivery: Система має зручний інтерфейс для користувачів і продавців. Користувач може здійснити замовлення онлайн або через мобільний додаток. Додаткові функції включають відстеження замовлення, керування доставкою та зворотній зв'язок.

− Nova Poshta: Система також пропонує онлайн замовлення через веб-сайт або мобільний додаток. Вона також має функції відстеження та управління замовленням.

− Укрпошта: Укрпошта надає можливість замовлення доставки онлайн, але її інтерфейс може бути менш інтуїтивно зрозумілим порівняно з іншими сервісами.

### 2. Управління відправленням:

− Delivery: Сервіс надає інструменти для відправників та одержувачів для керування відправленням. Це включає можливість зміни адреси доставки, часу або вибору інших параметрів.

− Nova Poshta: Nova Poshta пропонує аналогічні функції управління відправленням, які дозволяють користувачам контролювати процес доставки.

− Укрпошта: Управління відправленням також доступне для користувачів Укрпошти, але може бути обмежене в порівнянні з іншими сервісами.

### 3. Система відстеження:

− Delivery: Сервіс пропонує детальну систему відстеження, яка дозволяє користувачам в реальному часі переглядати місцезнаходження свого відправлення.

− Nova Poshta: Nova Poshta також має ефективну систему відстеження з можливістю дізнатися статус відправлення в будь-який момент.

− Укрпошта: Укрпошта також надає систему відстеження, але вона може бути менш деталізованою та оновлюватися рідше.

### 4. Послуги для бізнесу:

− Delivery: Delivery надає послуги для бізнесу, включаючи API для інтеграції з веб-сайтами та програмними продуктами.

− Nova Poshta: Nova Poshta також пропонує API та інші інструменти для підтримки бізнесу та електронної комерції.

− Укрпошта: Укрпошта також має певні послуги для бізнесу, але їхні можливості можуть бути обмежені.

Отже, загально взято, всі три сервіси - Delivery, Nova Poshta та Укрпошта - мають подібний функціонал, але можуть відрізнятися за ступенем зручності користування, доступністю певних опцій та підтримкою для бізнесу.

## Mind Map для заданої предметної області

1. Автентифікація та авторизація:

− Система автентифікації користувачів з різними ролями: адміністратор відділення, адміністратор системи, клієнт.

− Належність доступу до функціоналу залежно від ролі користувача.

2. Керування посилками:

− Можливість додавання, редагування та видалення вантажів з системи.

− Система відстеження вантажів для клієнтів та працівників.

3. Маршрутизація та доставка:

− Створення маршрутів для ефективної доставки вантажів.

− Можливість розрахунку часу та вартості доставки та інформування клієнтів про цей час.

4. Управління користувачами:

− Можливість керування ролями та правами доступу користувачів.

− Система управління користувачами для створення та видалення облікових записів.

Ці функціональні вимоги створюють основу для розробки програми для логістичних послуг, дозволяючи забезпечити високий рівень сервісу та ефективність управління логістичними процесами.

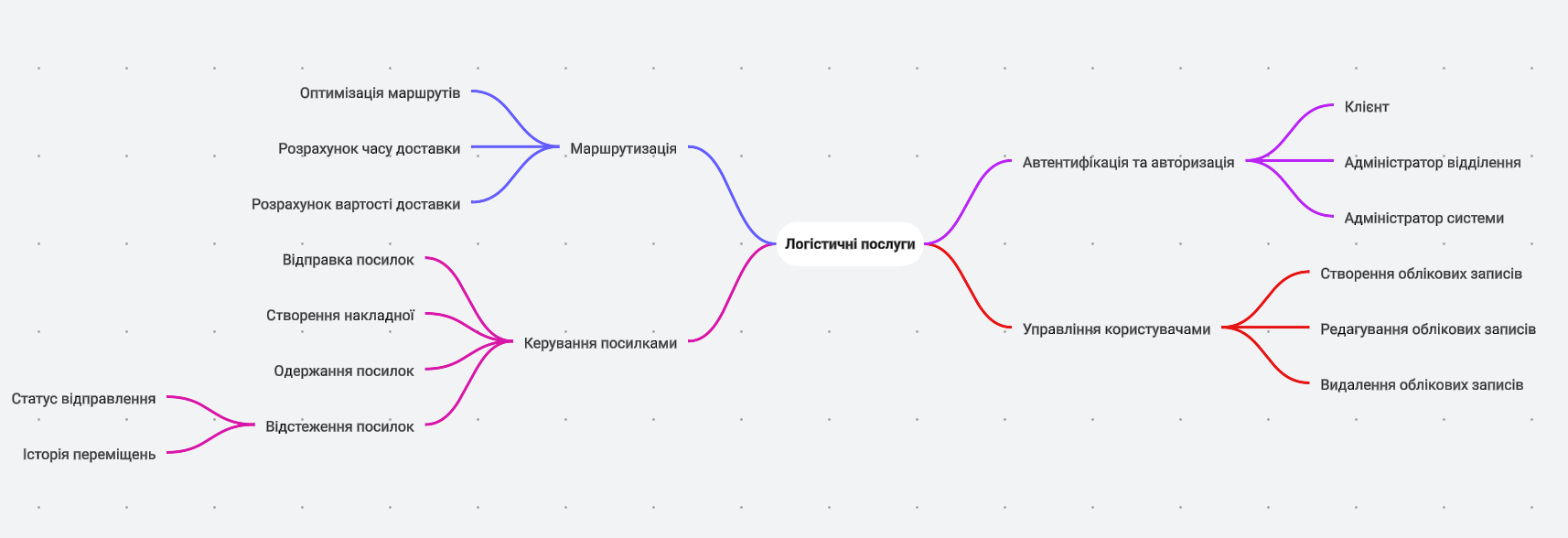


Рисунок 1.1 − Mind Map для програми Логістичних послуг

## Ролі користувачів

Враховуючи задану предметну область, можемо виділити три типи користувачів:

* Авторизований користувач (клієнт),
* Адміністратор відділення,
* Адміністратор системи.

Авторизований користувач оплачує посилки, формує повернення, відстежує відправлення, створює накладні, замовляє кур’єрів та перевіряє статус відправлень та історію переміщень.

Адміністратор відділення відповідає за інтеграцію посилок, сортування відправлень за напрямками, прийом та видачу посилок клієнтам.

Адміністратор системи додає нові відділення, налаштовує маршрути та інші системні параметри.

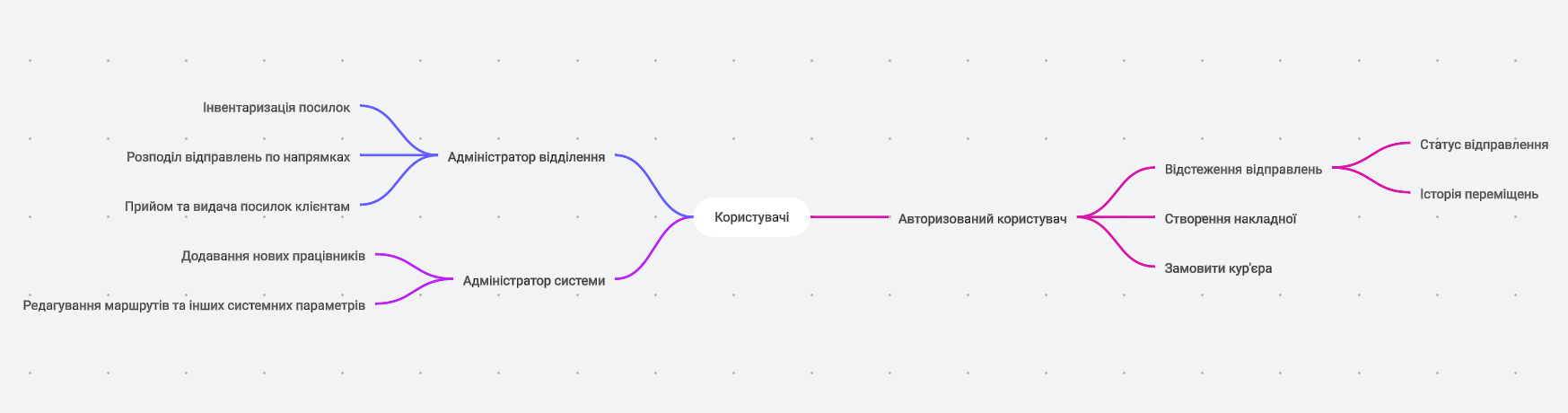


Рис. 1.2 − Mind Map користувачів програми

## Функціональні вимоги

Було виділено три типи користувачів: авторизований користувач (клієнт), адміністратор відділення та адміністратор системи.

Опишемо функціональні вимоги кожного типу користувачів.

Таблиця 1.1 – Функціональні вимоги користувача з роллю «Авторизований користувач»

|  |  |
| --- | --- |
| **Ідентифікатор** | **Функціональні вимоги** |
| АК.1. | Авторизація/Реєстрація в застосунку |
| АК.2. | Створити накладну на посилку  Примітка: статус накладної буде “створена” доки сама посилка не буде знаходитись у відділенні, і працівник не змінить її статус на “Підтверджено” |
| АК.3. | Відстежити посилку |
| АК.4. | Переглянути особистий профіль та відредагувати його |
| АК.5. | Переглянути архів посилок |

Таблиця 1.2 – Функціональні вимоги користувача з роллю «Адміністратор відділення»

|  |  |
| --- | --- |
| **Ідентифікатор** | **Функціональні вимоги** |
| АВ.1. | Авторизація в застосунку |
| АВ.2. | Створити накладну на посилку/ Змінити статус існуючої накладної |
| АВ.3. | Видати посилку |
| АВ.4. | Розподілити посилки по напрямкам |

Таблиця 1.3 – Функціональні вимоги користувача з роллю «Адміністратор системи»

|  |  |
| --- | --- |
| **Ідентифікатор** | **Функціональні вимоги** |
| АС.1. | Авторизація в застосунку |
| АС.2. | Додавати/Редагувати/Видаляти працівників |
| АС.3. | Додавати/Редагувати/Видаляти маршрути |

## 

## Інтерфейс користувача та опис функціоналу

### Авторизація та реєстрація

Перше що ми бачимо при вході в програму після завантаження це логотип, назва, та кнопку яка переведе нас на сторінку авторизації за номером телефону. Також є кнопка “Don’t have an account?”, яка переведе нас на сторінку реєстрації.

Реєстрація створює акаунт клієнта. Створенням акаунту для працівника буде займатись адміністратор системи. Працівник може перемикати інтерфейс з “робочого” на “клієнтський” у своєму профілі, і навпаки.

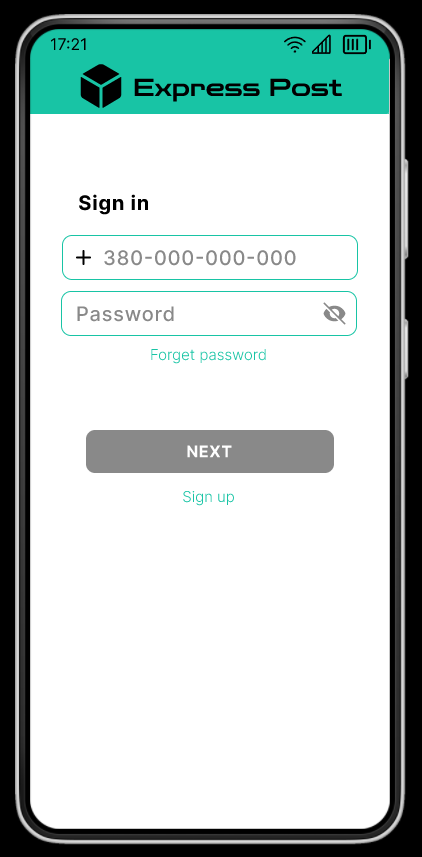
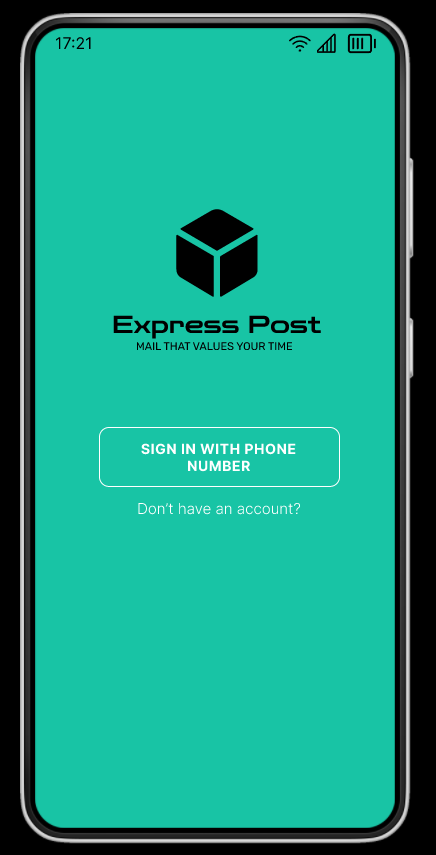


Рис. 3, 4 - Перший екран та авторизація

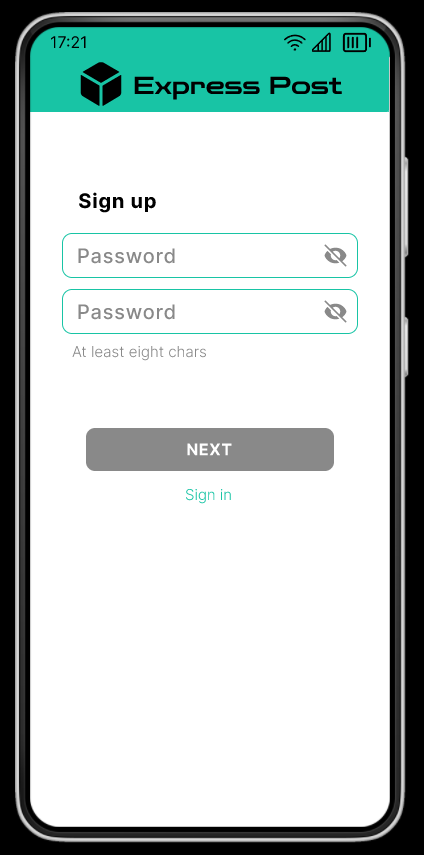
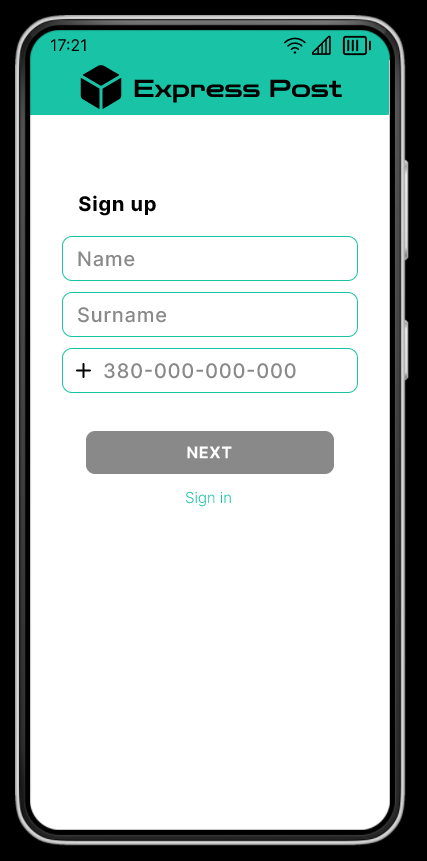


Рис. 5, 6 - Реєстрація

### Клієнтський інтерфейс

На екрані головного меню в верхній панелі екрану є кнопки які відкривають власний профіль та налаштування.

Більшу частину екрана займають чотири великих елементи які перенаправляють користувача на інші сторінки, а саме:

* “Мої посилки” - показує ті що вже прямують до клієнта, а також ті що вже прибули впродовж останнього тижня.
* “Створити посилку” - користувач може сам створити накладну указавши потрібні дані. Статус такої накладної буде “створено”, і доки посилка не попаде у відділення, з якого вона має відправитись, її статус не зміниться на “Підтверджено”.
* “Відділення” - користувач може переглянути відділення пошти на мапі.
* “Замовити кур’єра” - клієнт може замовити кур’єра на конкретний день та годину, указавши який товар має бути доставлений у відділення пошти.

Також є функція “Відслідкувати посилку” за номером накладної, яка розташована в верхній частині екрану.

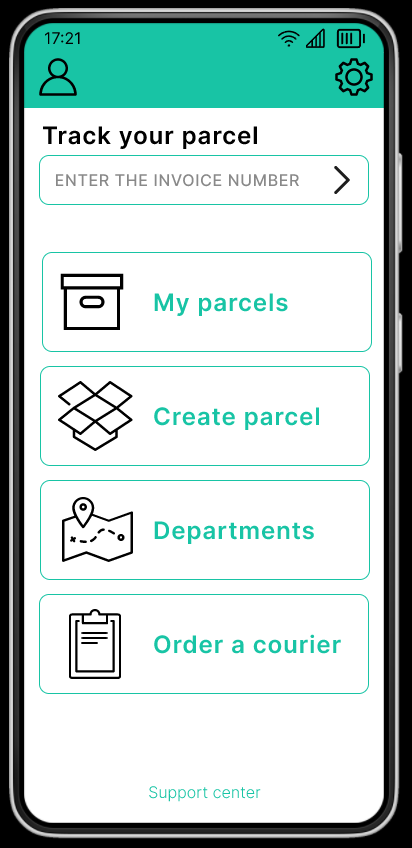


Рис. 6 - Головне меню

При відслідковуванні посилки за номером накладної ми побачимо повідомлення “Не знайдено жодної посилки”, або побачимо одну посилку але з мінімальною інформацією, а саме статус посилки та міста звідки вона їде і куди.

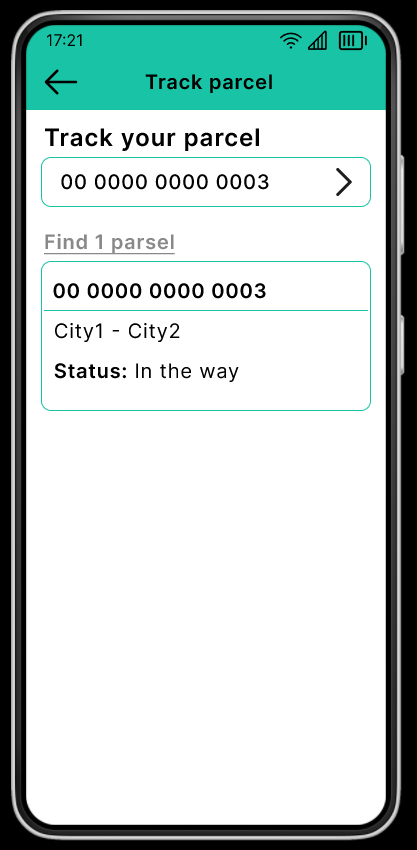


Рис 7 - Відслідкувати посилку

Відкриваючи “Мої посилки” можемо побачити коротку інформацію про них, також в верхній частині екрану є кнопка, натиснувши на яку на екрані будуть показані три додаткові дії: “Відслідкувати посилку”, “Створити посилку”, “Замовити кур’єра”.

Натиснувши на одну з посилок можна відкрити більш детальну інформацію про неї. Наприклад дату коди була відправлена, та дата коли має прибути, а також адреса куди прибуде. Також можна подивитись деталі маршруту: проміжні міста та відділення між місцем відправки та пунктом призначення, а також час коли прибуде/виїде. А також є кнопка для оплати доставки в самій програмі а не у відділенні.

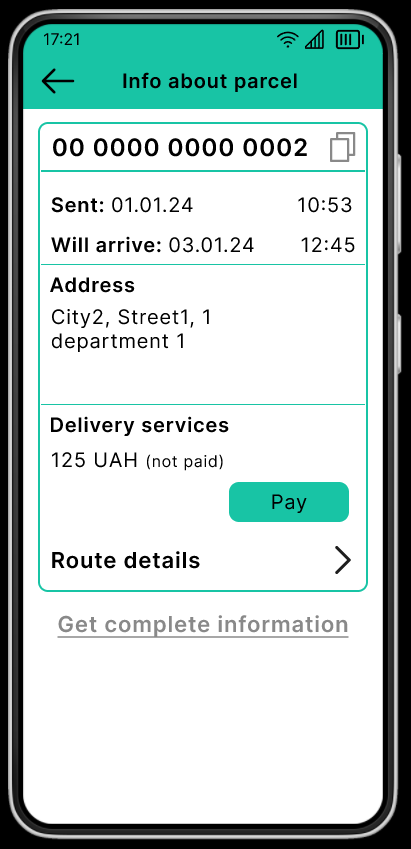
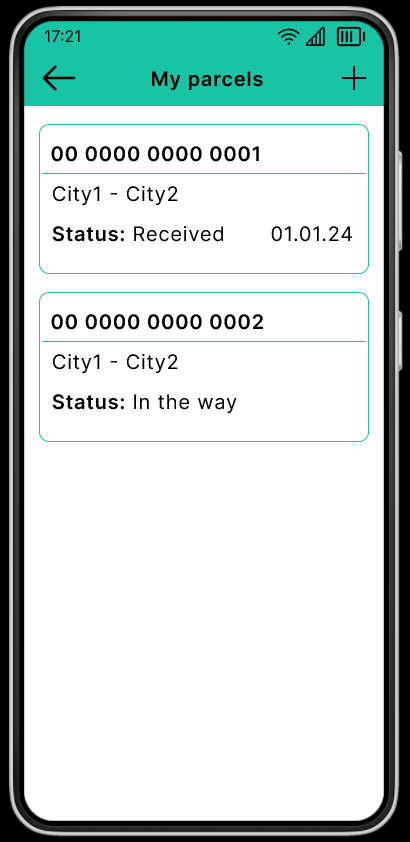


Рис 8, 9 - Мої посилки та інформація

Сторінка для створення накладної умовно поділена на дві частини (рис.10): інформація про відправника (власник акаунту, інформація окрім адреси заповнюється автоматично) та інформація про отримувача. Для кожної з “ролей” треба ввести прізвище ім’я, номер телефону, також обрати міста, адреси початкового та кінцевого пункту призначення (можна обрати відділення або за адресою).

На наступному етапі (рис.11) треба ввести інформацію про саму посилку а саме тип (обрати з випадаючого списку: документи / посилка до 2 кг / посилка до 5 кг і тд.), кількість посилок такого типу та оголошена вартість (якщо через непередбачувану ситуацію посилка буде втрачена її вартість компенсують). Також є кнопка для розрахунку вартості.

В верхній панелі екрану є кнопка яка додасть ще один блок для посилки. Ця функція додана якщо треба доставити декілька посилок різного розміру одному отримувачу.

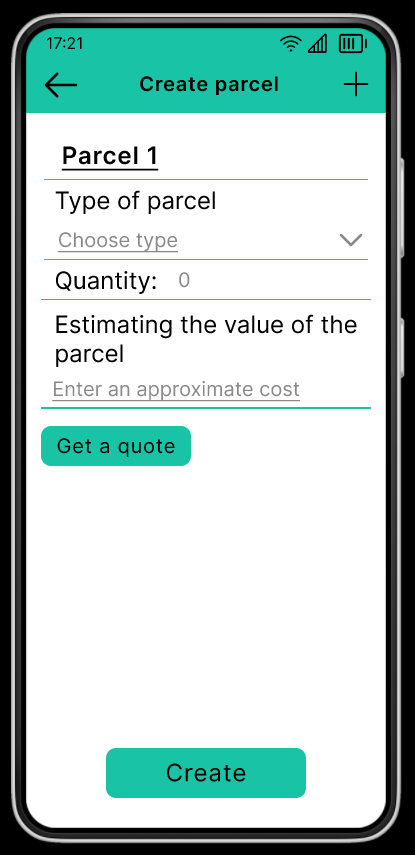
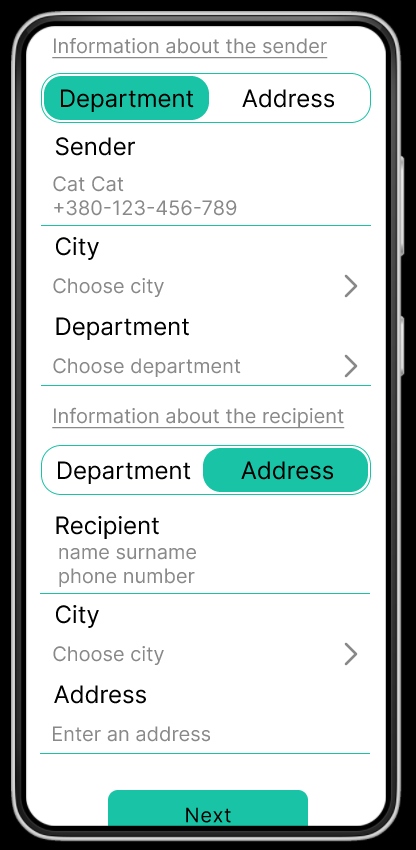


Рис 10, 11 - Створити накладну

Після створення накладної вона буде відображатись у власному профілі вкладці “Створені посилки” доки посилка не буде доставлена у відділення де її статус зміниться, і відображатись буде у “Мої посилки”.

Також однією з немало важливих функцій є викликати кур’єра (рис.13). Інформація про клієнта така сама як і при створенні накладної у розділі “Інформація про відправника”: прізвище, ім’я, номер телефону, місто, адреса. Також треба ввести дату та час який вам зручно щоб кур’єр прийшов. І також має бути указана деяка інформація про посилку, а саме її тип кількість, та хто зі сторін (відправник / отримувач) сплачує послуги доставки.

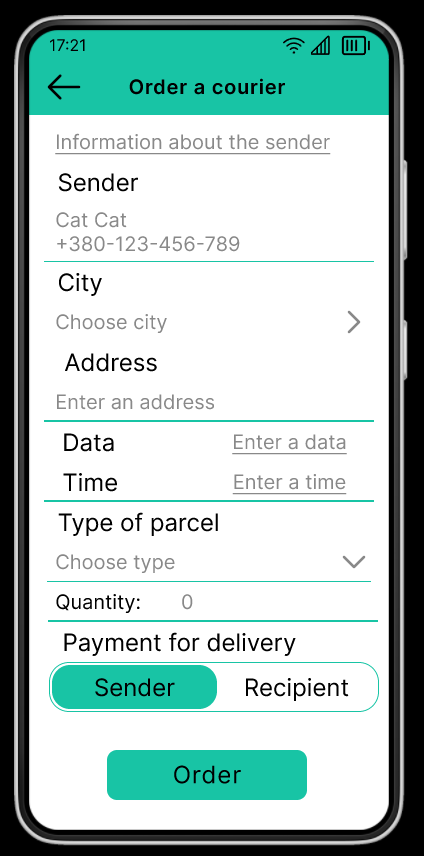
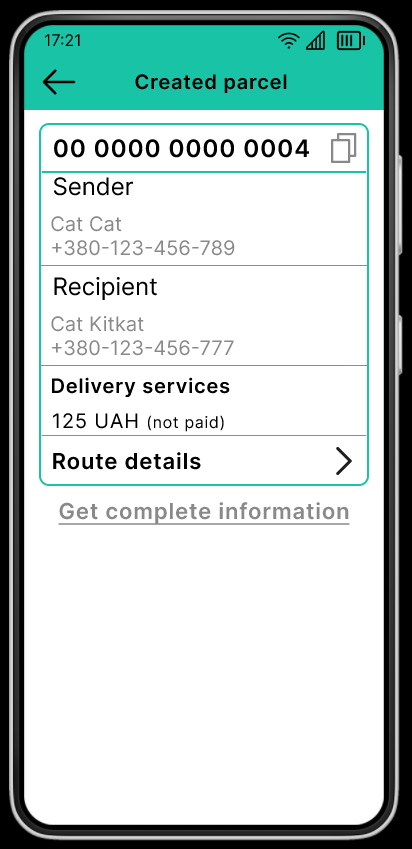


Рис 12, 13 - Створена накладна; форма замовлення кур’єра

Враховуючи специфіку заданої предметної області, вид моєї роботи та складнощі в реалізації деяких функцій було створено сторінку (рис.14) із вибаченнями за незручності та повідомленням що “ця функція буде додана пізніше”. До таких функцій які не будуть реалізовані відносяться:

* Перегляд відділень пошти на карті;
* Оплата доставки.

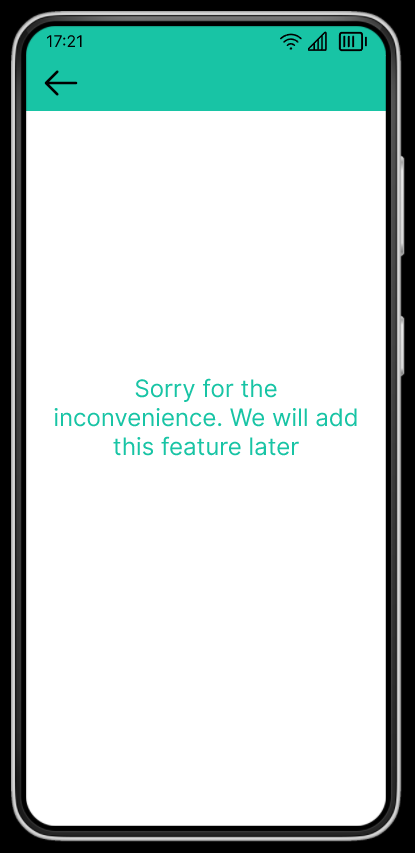


Рис 14 - Повідомлення

### Інтерфейс адміністратора відділення

### Інтерфейс адміністратора системи

## Функціональні тести

### Функціональні тести для ролі користувача “Авторизований користувач”

На основі опису предметної області, а також функціональних вимог і екранних форм були розроблені функціональні тести для ролі користувача «Авторизований користувач» (див. табл. 1.4).

Таблиця 1.4 – Функціональні тести для ролі користувача «Авторизований користувач»

|  |  |  |
| --- | --- | --- |
| **Іденти-фікатор тесту** | **Послідовність дій користувача** | **Очікуваний результат** |
| АК.1 Клієнт може авторизуватись/зареєструватись в застосунку | | |
| АК.1.1 | 1. Відкрити програму 2. Натиснути кнопку “Ввійти за допомогою номеру телефону” 3. Ввести номер телефону: “380123456789” 4. Ввести Пароль: “пароль123” 5. Натиснути кнопку “Увійти” | Успішна авторизація |
| АК.1.2 | 1. Відкрити програму 2. Натиснути кнопку “Не має аккаунту” 3. Ввести ім’я: “Катерина” 4. Ввести прізвище: “Кіт” 5. Натиснути кнопку “Далі” 6. Ввести пароль: “пароль123” 7. Вдруге ввести пароль: “пароль123” 8. Натиснути кнопку “Зареєстуватися” | Успішна реєстрація |
| АК.1.3 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Таблиця 1.4 – Функціональні тести для ролі користувача «Адміністратор системи»

|  |  |  |
| --- | --- | --- |
| **Іденти-фікатор тесту** | **Послідовність дій користувача** | **Очікуваний результат** |
| АК.1 Клієнт може авторизуватись/зареєструватись в застосунку | | |
| АК.1.1 | 1. Відкрити програму 2. Натиснути кнопку “Ввійти за допомогою номеру телефону” 3. Ввести номер телефону: “380123456789” 4. Ввести Пароль: “пароль123” 5. Натиснути кнопку “Увійти” | Успішна авторизація |
| АК.1.2 | 1. Відкрити програму 2. Натиснути кнопку “Не має аккаунту” 3. Ввести ім’я: “Катерина” 4. Ввести прізвище: “Кіт” 5. Натиснути кнопку “Далі” 6. Ввести пароль: “пароль123” 7. Вдруге ввести пароль: “пароль123” 8. Натиснути кнопку “Зареєстуватися” | Успішна реєстрація |
| АК.1.3 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Таблиця 1.4 – Функціональні тести для ролі користувача «Адміністратор відділення»

|  |  |  |
| --- | --- | --- |
| **Іденти-фікатор тесту** | **Послідовність дій користувача** | **Очікуваний результат** |
| АК.1 Клієнт може авторизуватись/зареєструватись в застосунку | | |
| АК.1.1 | 1. Відкрити програму 2. Натиснути кнопку “Ввійти за допомогою номеру телефону” 3. Ввести номер телефону: “380123456789” 4. Ввести Пароль: “пароль123” 5. Натиснути кнопку “Увійти” | Успішна авторизація |
| АК.1.2 | 1. Відкрити програму 2. Натиснути кнопку “Не має аккаунту” 3. Ввести ім’я: “Катерина” 4. Ввести прізвище: “Кіт” 5. Натиснути кнопку “Далі” 6. Ввести пароль: “пароль123” 7. Вдруге ввести пароль: “пароль123” 8. Натиснути кнопку “Зареєстуватися” | Успішна реєстрація |
| АК.1.3 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 2. Побудова моделі предметної області у вигляді діаграми класів

## Класи предметної області та їх характеристики

**Клас «User» – користувач**

Таблиця 2.1 – Виділення характеристик класу «User»

|  |  |  |
| --- | --- | --- |
| **Назва характеристики** | **Тип значення** | **Обмеження** |
| ID | INT | Не пустий рядок, наявність символу ‘@’ |
| FirstName | Varchar | Не пустий рядок, довжина: хоча б 2 символ |
| LastName | Varchar | Не пустий рядок, довжина: хоча б 2 символ |
| PhoneNumber | Char(10) | Не пустий рядок |
| Password | varchar | Не пустий рядок |
| Role | Enum | Клієнт/Адміністратор відділення/Адміністратор системи |

Таблиця 2.2 – Виділення поведінки класу «User»

|  |  |  |  |
| --- | --- | --- | --- |
| **Назва поведінки** | **Опис вхідних параметрів** | **Тип значення, що повертається** | **Бізнес-правила** |
| Оформити купівлю білетів | Фільм,  дата,  час,  кількість білетів | true/false – замовлення оформлено / помилка при оформленні замовлення;  номер замовлення  (у випадку true) | 1. Перевірити чи дійсні вхідні дані (фільм, дата, час). 2. Перевірити наявність необхідної кількості білетів. 3. Якщо пройдені перевірки, то оформити замовлення. |
| Повернути куплені білети | Номер замовлення | true/false – повернення оформлено / помилка при оформленні повернення. | 1. Перевірити чи дійсний номер замовлення. 2. Перевірити чи не відбувся вже сеанс. 3. Якщо до сеансу залишилося більше 24 годин, то повернути повну суму грошей, інакше – 50%. |
| … | … | … | … |

**Клас «Branch» – користувач**

Таблиця 2.1 – Виділення характеристик класу «User»

|  |  |  |
| --- | --- | --- |
| **Назва характеристики** | **Тип значення** | **Обмеження** |
| ID | INT | Не пустий рядок, наявність символу ‘@’ |
| FirstName | Varchar | Не пустий рядок, довжина: хоча б 2 символ |
| LastName | Varchar | Не пустий рядок, довжина: хоча б 2 символ |
| PhoneNumber | Char(10) | Не пустий рядок |
| Password | varchar | Не пустий рядок |
| Role | Enum | Клієнт/Адміністратор відділення/Адміністратор системи |

Таблиця 2.2 – Виділення поведінки класу «User»

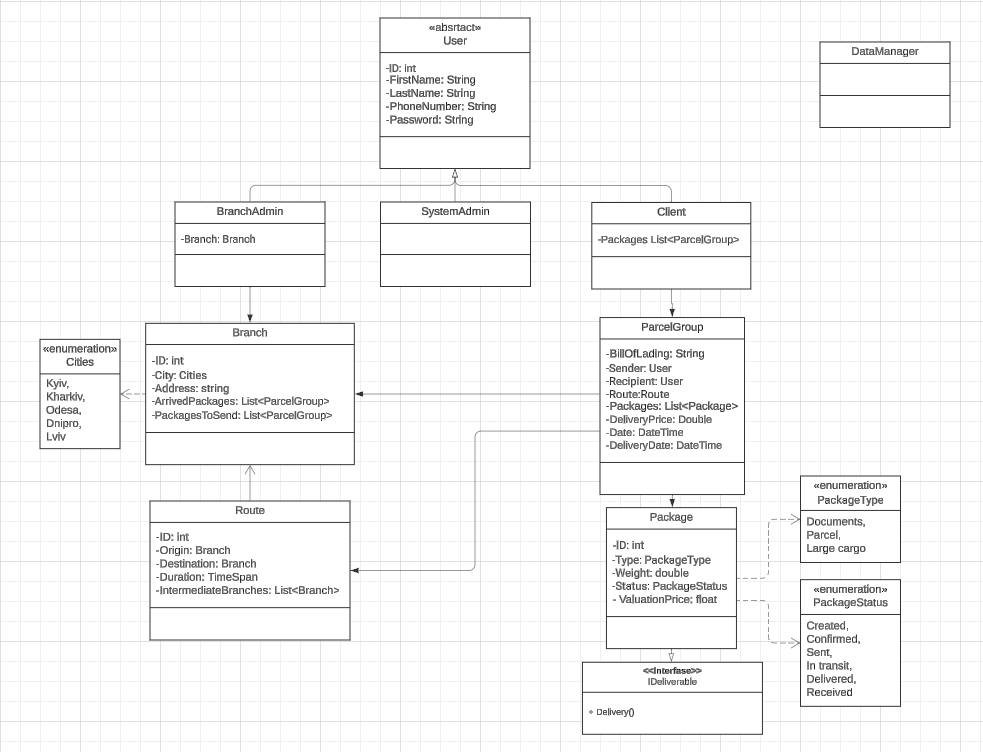


Рис. 2.1 - UML-Diagram