

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмних систем і технологій

Дисципліна
«Основи програмної інженерії»

Лабораторна робота № 4
«ВСТАНОВЛЕННЯ ВИМОГ ДО ФУНКЦІОНАЛЬНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»

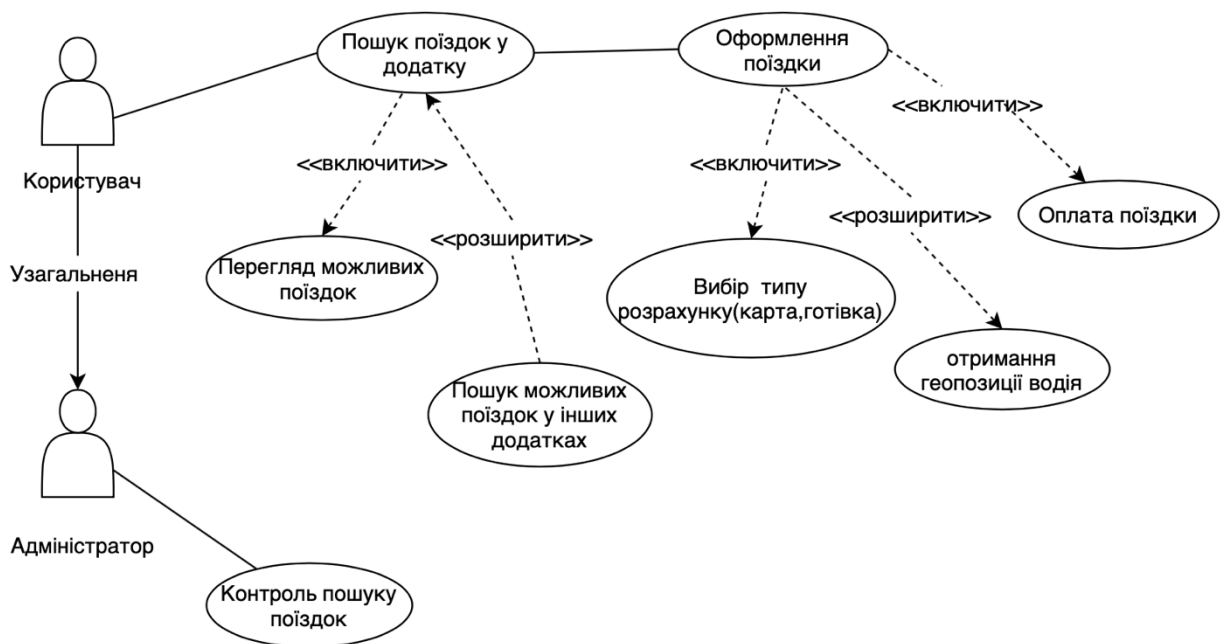
на тему:
«Автоматизована система продажу «Виклик таксі»»

Виконав:	Богатько Олександр Геннадійович	Перевірила:	Юрчук Ірина Аркаїївна
Група	ІПЗ-12	Дата перевірки	
Форма навчання	денна	Оцінка	
Спеціальність	121		
2020			

Завдання

1. За узгодженням з викладачем обрати варіант завдання (предметну галузь) для виконання лабораторних робіт.
2. Провести попередній аналіз предметної галузі, визначити функції ПЗ, що проектується.
3. Побудувати діаграму прецедентів на основі проведеного попереднього аналізу.
4. Для прецедентів, визначених у діаграмі прецедентів побудувати вербальні специфікації.

Діаграма прецедентів



Вербальні специфікації прецедентів

1. Пошук поїздок у додатку
 - 1) Короткий опис. Користувач заходить у додаток і здійснює пошук поїздок
 - 2) Суб'єкт – Користувач
 - 3) Передумова – користувач знаходить у додаток.
 - 4) Основний потік.

Користувач заходить додаток і вводить інформацію про поїздку – дату відправлення, місце відправлення місце призначення.

Якщо водіїв немає, виконується А1. Якщо додаток не завантажується, виконується А2.
 - 5) Альтернативні потоки

А1: Немає поряд водіїв. Користувач отримує повідомлення про те, що поряд немає водіїв.

А2: Не завантажується додаток. Користувач отримує повідомлення. Що потрібно почекати.
 - 6) Постумови

Якщо поблизу є вільні водії, користувач оформлює поїздку.

2. Перегляд можливих поїздок.

1) Короткий опис. Користувач здійснює перегляд всіх можливих поїздок для інформації яку вказав

2) Суб'єкт – Користувач

3) Передумова – Пошук поїздок у додатку.

4) Основний потік.

Користувач вибирає таксі, яке йому найкраще підходить із усіх, що відобразилися. Якщо вільних водіїв немає, виконується А1.

5) Альтернативні потоки

А1: Немає вільних водіїв. Користувач отримує повідомлення і пропозицію зачекати.

6) Постумови

Користувач оформлює поїздку.

3. Оформлення поїздки

1) Короткий опис. Користувач оформлює поїздку.

2) Суб'єкт – Користувач

3) Передумова – користувач обирає пункт призначення.

4) Основний потік.

Користувач оформляє обрану поїздку – заповнює форму, оплачує, отримує електронний чек. Якщо неможливо оформити поїздку, виконується А1.

5) Альтернативні потоки

А1: Неможливо оформити поїздку. Користувач отримує повідомлення про причину і пропозицію ще раз оформити поїздку.

6) Постумови

Користувач отримує геопозицію водія

4. Вибір типу розрахунку.

1) Короткий опис. Користувач вибирає вид таксі і починає оформлення: обирає тип розрахунку.

2) Суб'єкт – Користувач

3) Передумова – користувач обирає поїздку.

4) Основний потік.

Користувач заповнює форму, яка містить особисті дані для розрахунку(номер картки, або решта з готівки), та дані про поїздку. Після заповнення користувач відправляє форму і переходить до оплати. Якщо форма заповнена неправильно, виконується А1. Якщо форма заповнена не повністю, виконується А2.

5) Альтернативні потоки

А1: Неправильно заповнена форма. Користувач отримує повідомлення та ще раз заповнює неправильно заповнене поле.

А2: Не повністю заповнена форма. Користувач отримує повідомлення і заповнює форму до кінця.

6) Постумови

Користувач оплачує поїздку.

5. Оплата поїздки.

- 1) Короткий опис. Користувач отримує данні о поїзді, для якої заповнив форму
- 2) Суб'єкт – Користувач
- 3) Передумова – користувач заповнює форму
- 4) Основний потік.

Користувач вводить дані карти та одноразовий пароль з смс. Якщо оплата успішна, він отримує електронний варіант квитка. Якщо оплата неуспішна, виконується A1.

5) Альтернативні потоки

A1: Неуспішна оплата. Користувач отримує повідомлення та пропозицію спробувати оплатити ще раз.

6) Постумови

Користувач отримує геопозицію водія.

6. Контроль пошуку поїздок.

- 1) Короткий опис. Адміністратор контролює правильність заповнення форм, вибору, оплати поїздки та її оформлення.
- 2) Суб'єкт – Адміністратор.
- 3) Передумова – немає.
- 4) Основний потік. Адміністратор контролює процеси, які відбуваються при замовленні поїздки. Якщо відбувається збій системи, відбувається A1.
- 5) Альтернативні потоки

A1: Збій системи. Відбувається резервне копіювання даних та перезавантаження системи.

6) Постумови

Немає.

7. Пошук можливих поїздок у інших додатках.

- 1) Короткий опис. Користувач отримує пропозицію замовити поїздку у іншому додатку.
- 2) Суб'єкт – Користувач
- 3) Передумова – користувач здійснює пошук водіїв.
- 4) Основний потік.

Користувач вводить дані, і якщо немає поблизу вільних водіїв, йому виводять пропозицію переглянути інші додатки. Якщо пропозицій немає, A1

5) Альтернативні потоки

A1: Немає пропозицій. Користувач отримує повідомлення.

6) Постумови

Користувач продовжує пошук або залишає додаток.

Висновок

В цій лабораторній роботі було встановлено функціональні вимоги до програмного забезпечення, яке розробляється – автоматизованої системи «Виклик таксі». Було побудовано графічну форму специфікації – діаграму прецедентів UML. Було визначено основні пункти, які повинні виконуватися в процесі роботи з системою. Було зроблено вербальний опис кожного прецеденту. Робота була виконана за попереднім аналізом предметної галузі.