Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра ЕОМ



Звіт

до лабораторної роботи №1

з дисципліни: “Інженерія програмного забезпечення”

на тему: «UML діаграми варіантів використання та UML діаграми активності.»

Варіант №15

Виконав:

ст.гр. КІ-34

Ольховик О.С.

Прийняв:

ст. викл. каф.

Цигилик Л.О.

**Львів 2022**

**Мета:** Освоїти принципи створення UML діаграм варіантів використання що описують сценарій роботи системи (Use case diagram) та діаграм активності для клієнтської та серверної частин.

**Завдання:** Розробити UML діаграми сценаріїв роботи клієнтської та серверної частин а також розробити діаграми активності для кожного сценарію. Кількість UML діаграм повинна бути не менше – 8.

**Варіант №15:** Система бронювання та купівлі авіаквитків.

1. **Клієнтська частина**

**1.1 UML діаграма сценарію роботи клієнтської частини**

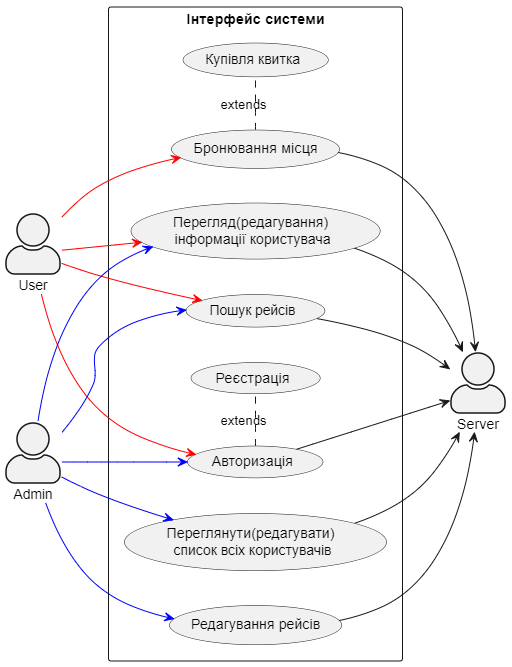
****

Рис. 1.1 Діаграма сценарію роботи для клієнтської частини.

**Опис діаграми клієнтської частини**

Дана діаграма зображає можливі сценарії роботи клієнтської частини. Оскільки система взаємодіє з користувачем, адміністратором та сервером, то на ній присутні три актори: User, Admin та Server. Операція входу в систему передбачає можливість реєстрації нового користувача, оскільки дана процедура не є обов’язковою, то на діаграмі вона позначена зв’язком розширення (extended). Кожний подальший варіант використання має прямий зв’язок з входом у систему та сервером. Передача даних можлива як від клієнта до сервера, так і від сервера до клієнта.

**1.2 Процес авторизації в систему**

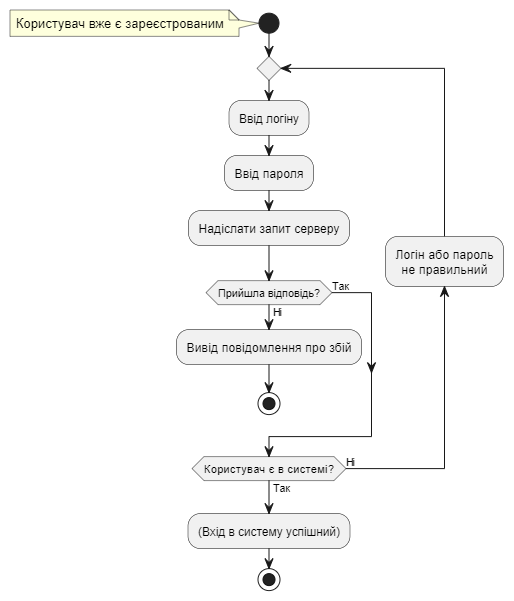
****

Рис. 1.2 Діаграма активності для авторизації.

**Опис діаграми**

Дана діаграма описує послідовність дій, які потрібно виконати для успішної авторизації в систему. Користувач повинен ввести свій логін та пароль. Якщо логін або пароль були введені не коректно, то користувач отримає повідомлення, що він зробив щось не так. Якщо ж дані були введені правильно, то користувачу буде дозволено зайти в додаток, де він зможе користуватись його функціоналом, залежно від того, ким він є – Адміністратором чи Клієнтом.

**1.3 Процес реєстрації в систему**

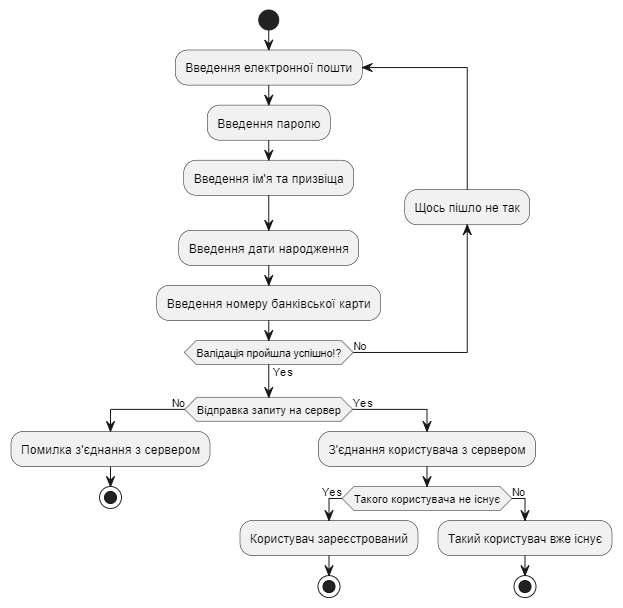
****

Рис. 1.3 Діаграма активності для реєстрації.

**Опис діаграми**

Якщо ж користувач виявився відсутнім в базі даних, то йому треба зареєструвати себе, ввівши потрібні для цього дані: ім’я, призвіще, електронну пошту, пароль та номер телефону. Зробивши це, дані відправляться на сервер, де пройдуть перевірку на коректність вводу. Після цього користувач отримає повідомлення про результат – або його зареєстровано, або виникли якісь помилки.

**1.4 Процес отримання даних користувача**

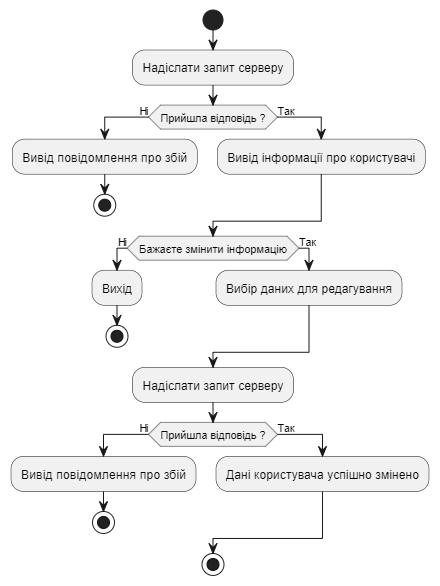
****

Рис. 1.4 Діаграма активності для отримання даних користувача.

**Опис діаграми**

Будь-який користувач може отримати записані дані до БД, які він вказав при реєстрації. Зробити він це може щоб впевнитися в тому, що вказані вони були правильно, якщо ж ні, то він має можливість їх редагувати. Для редагування доступні пароль, ім’я, призвіще та номер телефону. Після змінення, нові дані відправляються на сервер, проходять перевірку і користувач отримує результат. Якщо дані подібні до попередніх, то користувач має вказати заново, якщо ж було зроблено все правильно, дані будуть оновлені.

**1.5 Процес отримання списку всіх користувачів**

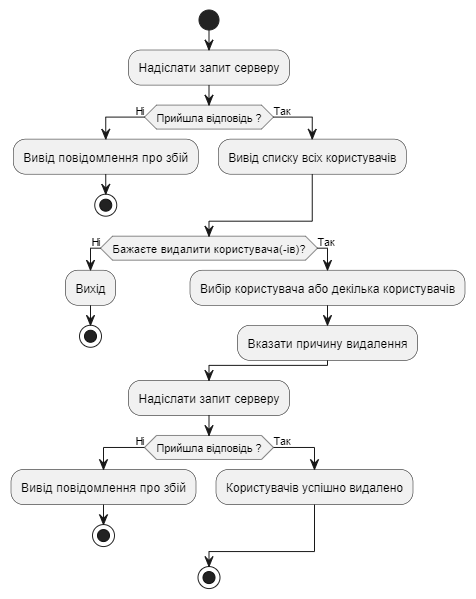
****

Рис. 1.5 Діаграма активності для отримання списку користувачів.

**Опис Діаграми**

Ця опція доступна лише для адміністратора. Він може запросити у сервера список всіх зареєстрованих користувачів. Запит відправляється на сервер, де він створює список відповідно до записів в БД. Адміністратор отримує цей список і може переглянути інформацію зареєстрованих користувачів. Також адміністратор має можливість видаляти користувачів з бази даних, проте має вказати причину, щоб користувач, який буде видалений, отримав сповіщення на його електрону пошту, що його було видалено, та чому.

**1.6 Процес отримання списку рейсів**

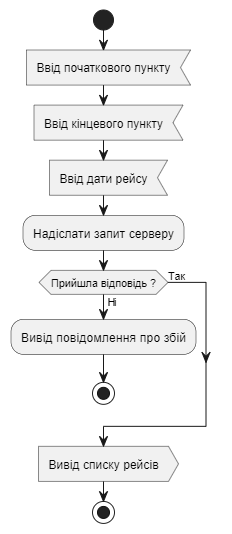
****

Рис. 1.6 Діаграма активності для отримання списку рейсів.

Щоб отримати список рейсів, користувач повинен вказати початкову точку - звідки планує відправлятись, та кінцеву – куди планує прибути. Також потрібно вказати дату відправлення. Після чого запит на отримання списку відправляється на сервер для оброблення. Якщо все було зроблено правильно і введені коректні дані, користувач отримає задовільний результат, в іншому, отримає повідомлення про помилку.

* 1. **Процес бронювання місця та купівлі квитка**

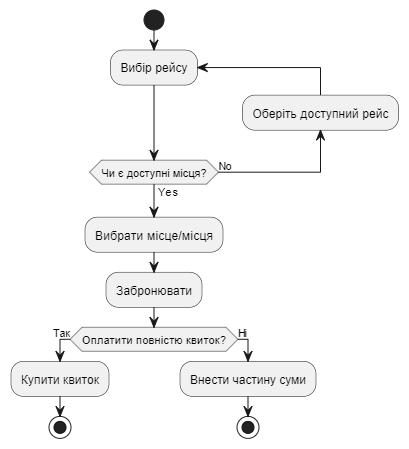
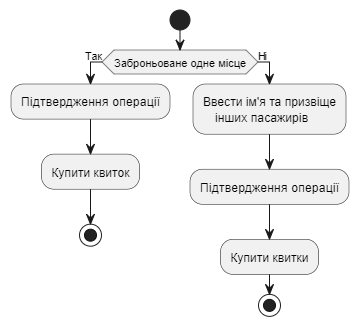
****

Рис. 1.7.1 Діаграма активності бронювання. Рис. 1.7.2 Діаграма активності купівлі квитка.

**Опис діаграм**

Після того, як користувач переглянув список рейсів, отриманих за його запитом, він може обрати будь-який, що підходить для нього і забронювати собі місце або декілька місць, залежно від кількості попутників. Забронювавши місце або місця, користувач може обрати або купити квиток одразу, або заплатити частину суми при бронюванні, і частину пізніше. Якщо ж був обраний перший варіант(купівлі квитка), користувач дані користувача беруться з бази даних, і якщо пасажир не один, то має вказати ім’я та прізвище його попутників. Після цього запит відправляється на сервер і згодом користувач отримає від нього відповідь про успішне замовлення, або помилку.

1. **Серверна частина**

**2.1 UML діаграма сценарію роботи клієнтської частини**

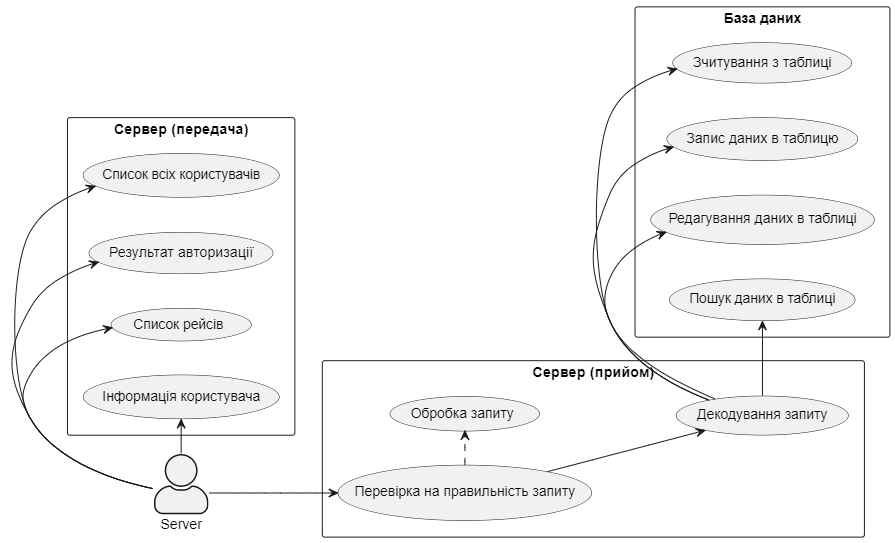
****

Рис. 2.1 Діаграма сценарію роботи для серверної частини.

**Опис діаграми сценаріїв серверної частини**

Дана діаграма зображає можливі сценарії роботи серверної частини. Оскільки сервер може взаємодіяти з базою даних і з клієнтом, ми маємо два основних розвитку подій. При спробі увійти до системи ідентифікується користувач в базі даних. Кожен пакет переданий від клієнта до сервера протоколом TCP/IP сервер перевіряє на цілісність та помилки і декодує.

**2.2 Процес авторизації та запису нових користувачів до БД**

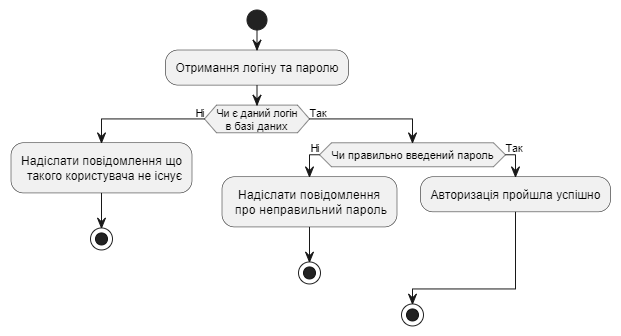
****

Рис. 2.2.1 Діаграма активності для авторизації.

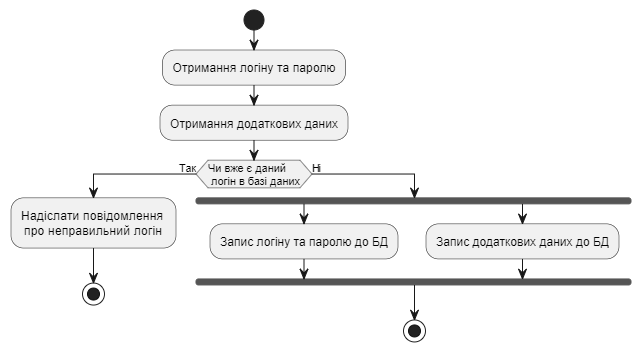
****

Рис. 2.2.2 Діаграма активності для запису нового користувача до БД.

**Опис діаграм**

На рис. 2.2.1. зображена діаграма активності для авторизації користувача в додаток. Сервер отримує запит від клієнтської частини, по якому він виконує пошук логіну, який вказав користувач, в базі даних. Якщо логін був знайдений, то після цього сервер вже співставляє введений пароль користувачем з тим, який записаний до бази даних в парі з логіном. У випадку коректності логіну та паролю, буде надіслане повідомлення користувачу про успішну авторизацію, інакше буде помилка. Якщо відповідний логін не був знайдений в базі даних, то користувачу буде запропоновано зареєструватись.

Даграма активності для реєстрацію зображена на рис 2.2.2. спочатку сервер отримує дані, які були відправлені користувачем через клієнтську частину, декодує їх та проводить перевірку, чи є вже подібний логін, вказаний при реєстрації, в базі даних. У випадку існування вже подібного логіну, користувачу буде надіслано повідомлення, що користувач з даним логіном вже зареєстрований. В інакшому випадку користувач отримає повідомлення про успішну реєстрацію.

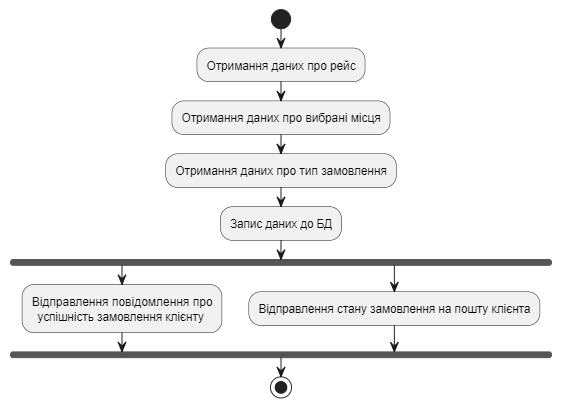
**2.3 Процес бронювання та купівлі квитка**

Рис. 2.3.1 Діаграма активності для бронювання.

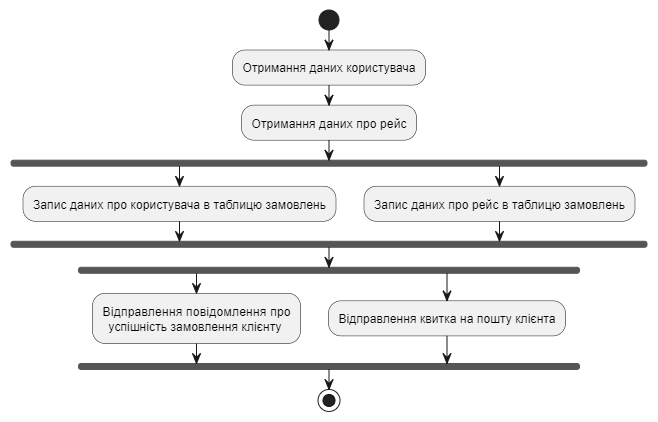
****

Рис. 2.3.2 Діаграма активності для купівлі квитка.

**Опис діаграм**

На рисунку 2.3.1 зображена діаграма активності для бронювання одного або декількох місць, залежно від кількості попутників зареєстрованого користувача. Вказані дані приходять на сервер, де він записує ті дані до БД, що місце/місця є заброньовані. При успішному записі, користувач отримає повідомлення в додаток та на пошту, де буде знаходитись детальна інформація щодо бронювання.

Якщо користувач вирішив купити квиток або квитки, то на сервер додатково приходять дані про вже про пасажира, а також можливих попутників(Рис. 2.3.2). У цьому випадку до БД вже записується, що на заброньовані місця куплені квитки.

**2.4 Процес відправки списку зареєстрованих користувачів або інфорації про авторизованого користувача**

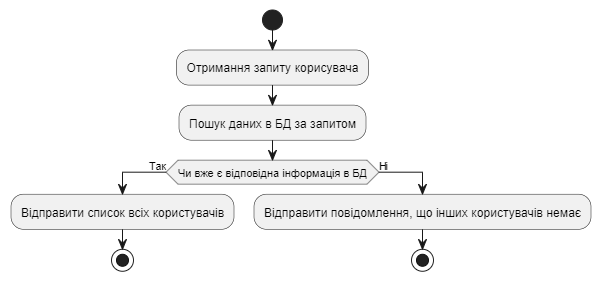
****

Рис. 2.4.1 Діаграма активності для пошуку списку користувачів в БД.

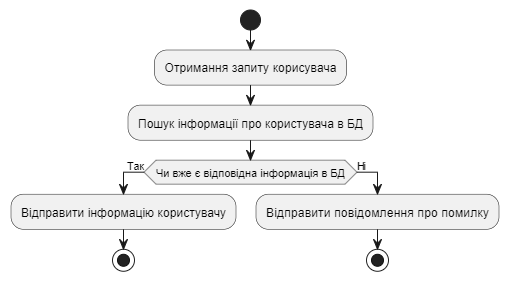
****

Рис. 2.4.2 Діаграма активності для пошуку інформації про користувача.

**Опис діаграм**

На рис 2.4.1 зображена діаграма активності для пошуку списку зареєстрованих користувачів в базі даних. Сервер отримує запит і за ним виконує пошук в БД, якщо на даний момент немає жодного зареєстрованого користувача, окрім адміністратора, то він отримає про це повідомлення, в іншому разі, буде надісланий список користувачів клієнту.

Також сервер може отримувати запит на пошук інформації про користувача, який виконує цей запит(Рис 2.4.2). Отримавши запит, сервер виконує пошук по БД, отримує потрібну інформацію і відправляє її користувачу, якщо ж виник збій, то користувач буде проінформований, що щось пішло не так.

**2.5 Процес відправки списку рейсів за отриманим маршрутом**

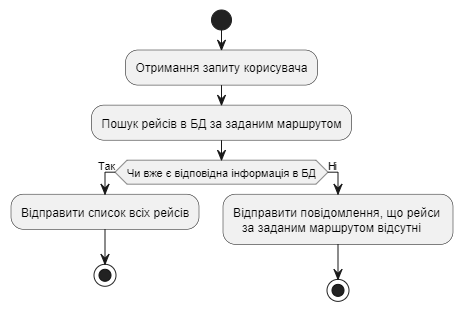
****

Рис. 2.5 Діаграма активності для пошуку рейсів в БД.

**Опис діаграми**

На рис 2.4.1 зображена діаграма активності для пошуку рейсів за заданим початковим і кінцевим пунктом призначення в базі даних. Сервер отримує запит і за ним виконує пошук в БД, якщо між вибраними точками відсутні рейси, користувач отримає відповідне повідомлення, а якщо ж будуть заплановані рейси, тоді користувачу буде надісланий список з відповідними рейсами.

**Висновок:** в результаті виконання даної лабораторної роботи №1 я освоїв принципи створення UML діаграм варіантів використання що описують сценарій роботи системи (Use case diagram) та діаграм активності для клієнтської та серверної частин.