Лабораторна робота №2 Написання UseCases

ЗАВДАННЯ

Відповідно до обраної предметої області на основі проведеного аналізу вимог замовника (лабораторна робота №1) розробити три різні usecases (по одному в короткій, поверхневій та повній формах відповідно) для свого проекту. Повна форма опису має містити всі пункти наведені в таблиці 1. Головний успішний сценарій повинен мати не менше 10 кроків. Передбачити не менше 5 альтернативних сценаріїв.

1.Створення посту на форумі

**Коротка форма:**  
Користувач входить в акаунт та переходить на сторінку створення посту. Користувач визначає мітку для посту та заповнює всю необхіду інформацію, система перевіряє заповнення полів та виводить повідомлення про підтвердження публікації посту. Користувач тисне так, система публікує пост.

**Поверхнева форма:**Головний сценарій (успішний):   
Користувач входить в акаунт(заповнюючи логін та пароль) та переходить на сторінку створення посту. Користувач визначає мітку для посту та заповнює всю необхіду інформацію, система перевіряє заповнення полів та виводить повідомлення про підтвердження публікації посту. Користувач тисне так, система публікує пост.

Альтернативні сценарії:

1. Всі необхідні поля не заповнені, система повідомляє про це користувача та просить їх заповнити
2. Збір системи, система виводить помилку та просить спробувати пізніше, звіт про помилку відправляється адміністратору.

**Повна:**

Створення посту на форумі  
**Scope:** Система організації “Пласт” (System)

**Level:** User-goal

**Primary Actor:** користувач

**Stakeholders and interests:**

1. Користувач: зацікавлений у швидкому створенні посту для швидкої та зручної комунікації користувачів

Preconditions: користувач увійшов в систему(авторизація)

**Main Success Scenario:**

1. Користувач входить в систему.

2. Користувач переходить на сторінку форуму.

3. Система відображає список існуючих тем та постів.

4. Користувач натискає кнопку «Створити новий пост».

5. Система відображає форму створення посту.

6. Користувач вводить заголовок посту.

7. Користувач вибирає мітку приналежності посту (наприклад, «Обговорення» чи «Питання»).

8. Користувач вводить текст посту.

9. Користувач натискає кнопку «Опублікувати».

10. Система перевіряє введені дані на коректність.

11. Система зберігає пост у базі даних.

12. Система відображає підтвердження про успішне створення посту.

13. Система оновлює список постів на форумі з новим постом.

**Extensions:**

1. Користувач не увійшов в систему:

* Система відображає форму входу/реєстрації.
* Користувач вводить облікові дані та входить в систему.

2. Користувач не заповнив обов’язкові поля:

* Система відображає повідомлення про необхідність заповнення обов’язкових полів.
* Користувач заповнює обов’язкові поля та повторює спробу.

3. Користувач вводить некоректний формат даних:

* Система відображає повідомлення про помилку формату даних.
* Користувач виправляє дані та повторює спробу.

4. Користувач хоче скасувати створення посту:

* Користувач натискає кнопку «Скасувати».
* Система скасовує створення посту та повертає користувача на сторінку форуму.

5. Система не може зберегти пост через технічні проблеми:

* Система відображає повідомлення про помилку збереження.
* Користувач може повторити спробу пізніше.

6. Користувач хоче попередньо переглянути пост:

* Користувач натискає кнопку «Попередній перегляд».
* Система відображає пост у режимі попереднього перегляду.
* Користувач продовжує редагування або опубліковує пост.

7. Користувач перевищує допустиму довжину тексту:

* Система відображає повідомлення про перевищення допустимої довжини.
* Користувач скорочує текст та повторює спробу.

**Special Requirements:**

Інтерфейс має бути зручним для користувачів з різним рівнем технічної підготовки.

Система повинна мати можливість інтеграції з поштовими сервісами для сповіщень про нові пости та коментарі.

**Technology and Data Variations List:**

Користувач може вводити дані через веб-інтерфейс.

Система може підтримувати кілька мов для зручності користувачів.

**Frequency of Occurrence:**

Нижче середнього

**Miscellaneous (Open Issues):**

Аналіз безпеки даних користувачів при створенні та збереженні постів.

Розробка механізму модерації постів для запобігання спаму та небажаного контенту.

Впровадження можливості редагування постів після їх публікації.

2. Редагування електронної пошти в профілі

**Коротка форма:**

Користувач входить в акаунт та переходить в профіль. Користувач   
вводить свій пароль для підтвердження особи, система звіряє дані та впускає користувача. Користувач бачить всі свої дані та переходить до своє електронної пошти. Користувач змінює свою електронну пошту, користувач тисне кнопку зберегти зміни, система просить підтвердження змін. Користувач тисне так, система зберігає їх

**Поверхнева форма:**

Головний сценарій (успішний):

Користувач входить в акаунт та переходить в профіль. Користувач

вводить свій пароль для підтвердження особи, система звіряє дані та впускає користувача. Користувач бачить всі свої дані та переходить до своє електронної пошти. Користувач змінює свою електронну пошту, користувач тисне кнопку зберегти зміни, система просить підтвердження змін. Користувач тисне так, система зберігає їх

Альтернативні сценарії:

1. Введений пароль неправильний, система повідомляє користувача та просить його ввести його ще раз.
2. Користувач вирішує не зберігати зміни, система повертає його в профіль
3. Збір системи, система виводить помилку та просить спробувати пізніше, звіт про помилку відправляється адміністратору.

**Повна**:

Редагування електронної пошти   
**Scope:** Система організації “Пласт” (System)

**Level:** User-goal

**Primary Actor:** користувач

**Stakeholders and interests:**

Користувач: Зацікавлений у можливості змінювати свої контактні дані для отримання актуальних сповіщень.

Адміністратори: Зацікавлені в актуальності контактної інформації для ефективної комунікації з користувачами.

**Preconditions**: користувач увійшов в систему(авторизація)

**Main Success Scenario:**

1. Користувач входить в систему.

2. Користувач переходить до свого профілю.

3. Система відображає дані профілю користувача.

4. Користувач натискає на кнопку «Редагувати профіль».

5. Система відображає форму редагування профілю з поточними даними.

6. Користувач змінює електронну пошту.

7. Користувач натискає кнопку «Зберегти зміни».

8. Система перевіряє валідність нової електронної пошти.

9. Система зберігає нову електронну пошту в базі даних.

10. Система надсилає підтвердження на нову електронну пошту.

11. Користувач отримує підтвердження про успішне редагування електронної пошти.

**Extensions:**

1. Користувач не увійшов в систему:

* Система відображає форму входу/реєстрації.
* Користувач вводить облікові дані та входить в систему.

2. Користувач не вказав формат електронної пошти:

* Система відображає повідомлення про помилку формату.
* Користувач вводить коректний формат електронної пошти.

3. Електронна пошта вже зайнята:

* Система відображає повідомлення про зайняту електронну пошту.
* Користувач вводить іншу електронну пошту.

4. Система не може зберегти зміни через технічні проблеми:

* Система відображає повідомлення про помилку збереження.
* Користувач може повторити спробу пізніше.

5. Користувач хоче скасувати зміни:

* Користувач натискає кнопку «Скасувати».
* Система відображає дані профілю без змін.

6. Користувач вводить неприпустимий символ у полі електронної пошти:

* Система відображає повідомлення про помилку неприпустимого символу.
* Користувач виправляє введення.

**Special Requirements:**

Інтерфейс має бути зручним для користувачів з різним рівнем технічної підготовки.

Система повинна мати можливість інтеграції з поштовими сервісами для верифікації електронної пошти.

Система повинна бути доступною 24/7.

**Technology and Data Variations List:**

Користувач може вводити дані через веб-інтерфейс

Система може підтримувати кілька мов для зручності користувачів.

**Frequency of Occurrence:**

Низька

**Miscellaneous (Open Issues):**

Аналіз безпеки даних користувачів при редагуванні контактної інформації.

Розробка механізму відновлення доступу до облікового запису у випадку зміни електронної пошти.

Впровадження двофакторної аутентифікації при зміні критичних даних профілю.

3. Створення заходу

**Коротка форма:**

Користувач входить в акаунт та переходить на сторінку створення заходу. Користувач заповнює всю необхіду інформацію, система перевіряє заповнення полів та виводить повідомлення про підтвердження створення заходу. Користувач тисне так, система публікує захід.

**Поверхнева форма:**

Головний сценарій (успішний):

Користувач входить в акаунт та переходить на сторінку створення заходу. Користувач заповнює всю необхіду інформацію, система перевіряє заповнення полів та виводить повідомлення про підтвердження створення заходу. Користувач тисне так, система публікує захід.

Альтернативні сценарії:

1. Організатор не заповнив всі обов’язкові поля, система відображає повідомлення про необхідність заповнення всіх обов’язкових полів.
2. Користувач вирішує не зберігати зміни, система повертає його до до переліку подій
3. Збір системи, система виводить помилку та просить спробувати пізніше, звіт про помилку відправляється адміністратору.

**Повна:**

Створення заходу  
**Scope:** Система організації “Пласт” (System)

**Level:** User-goal

**Primary Actor:** користувач

**Stakeholders and interests:**

Організатор: Зацікавлені у створенні та керуванні заходами.

Члени організації: Зацікавлені у відвідуванні та участі в заходах.

Керівництво організації: Зацікавлене у підвищенні залученості членів організації та підтриманні активності.

**Preconditions:** організатор увійшов в систему(авторизація)

**Main Success Scenario:**

1. Організатор входить у систему.

2. Організатор переходить на сторінку створення заходу.

3. Система відображає форму створення заходу.

4. Організатор вводить назву заходу.

5. Організатор вказує мету проведення заходу.

6. Організатор вводить час та дату початку і кінця заходу.

7. Організатор вказує місце проведення заходу.

8. Організатор вводить ім'я організатора заходу.

9. Організатор додає детальний опис заходу.

10. Організатор вибирає запрошених учасників із списку членів організації.

11. Організатор підтверджує створення заходу.

12. Система перевіряє введені дані на коректність.

13. Система зберігає новий захід у базі даних.

14. Система надсилає сповіщення запрошеним учасникам про новий захід.

15. Система відображає підтвердження створення заходу адміністратору або організатору.

**Extensions:**

1. Організатор не увійшов у систему:

* Система відображає форму входу.
* Організатор вводить облікові дані та входить у систему.

2. Організатор не заповнив всі обов’язкові поля:

* Система відображає повідомлення про необхідність заповнення всіх обов’язкових полів.
* Організатор заповнює обов’язкові поля та повторює спробу.

3. Система не може зберегти захід через технічні проблеми:

* Система відображає повідомлення про помилку збереження.
* Організатор може повторити спробу пізніше.

4. Організатор хоче скасувати створення заходу:

* Організатор натискає кнопку «Скасувати».
* Система скасовує процес створення заходу та повертає адміністратора або організатора на попередню сторінку.

5. Організатор додає коментарі до заходу:

* Організатор вводить коментарі до заходу (наприклад, особливі інструкції для учасників).
* Система додає коментарі до заходу.

6. Запрошені учасники не отримали сповіщення про захід:

* Система відображає журнал сповіщень.
* Організатор перевіряє статус сповіщень та може повторити відправку.

7. Захід конфліктує з іншим заходом за часом або місцем:

* Система відображає попередження про конфлікт.
* Організатор може змінити час або місце проведення заходу.

8. Запрошені учасники відповідають на сповіщення про захід:

* Система відображає відповіді учасників.
* Організатор переглядає відповіді та приймає відповідні рішення.

**Special Requirements:**

Система повинна забезпечувати безпеку та конфіденційність даних заходів та учасників.

Система повинна мати можливість інтеграції з іншими системами організації для сповіщень та звітів.

**Technology and Data Variations List:**

Організатор може вводити дані через веб-інтерфейс.

Система може підтримувати кілька мов для зручності користувачів

**Frequency of Occurrence:**

Середня

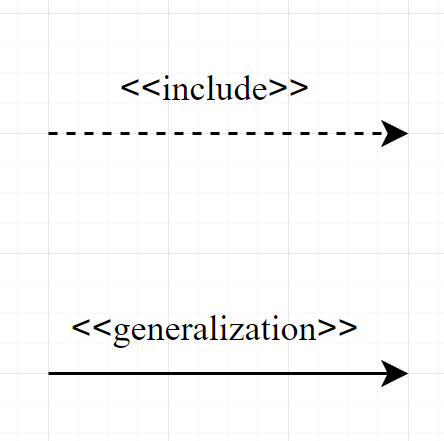
**Miscellaneous (Open Issues):**

Аналіз безпеки даних при створенні та збереженні заходів.

Розробка механізму обробки конфліктів між заходами.

Впровадження можливості автоматичного нагадування про майбутні заходи.

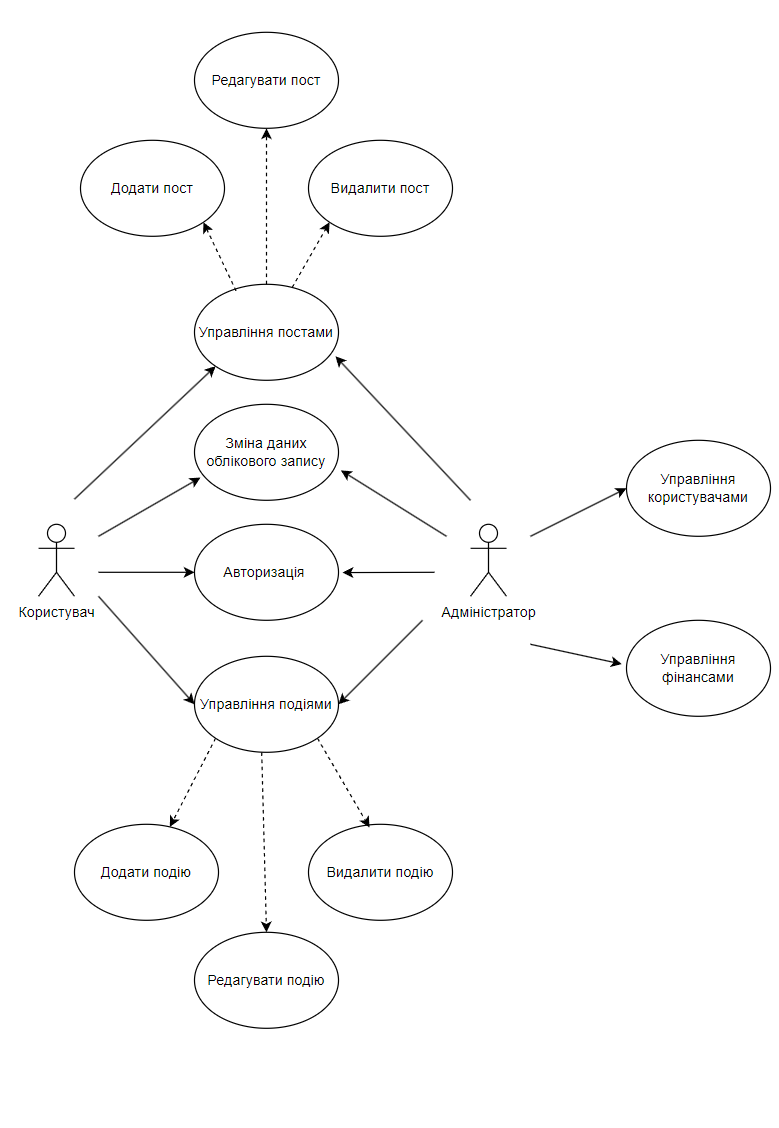
Використані види зв’язків:

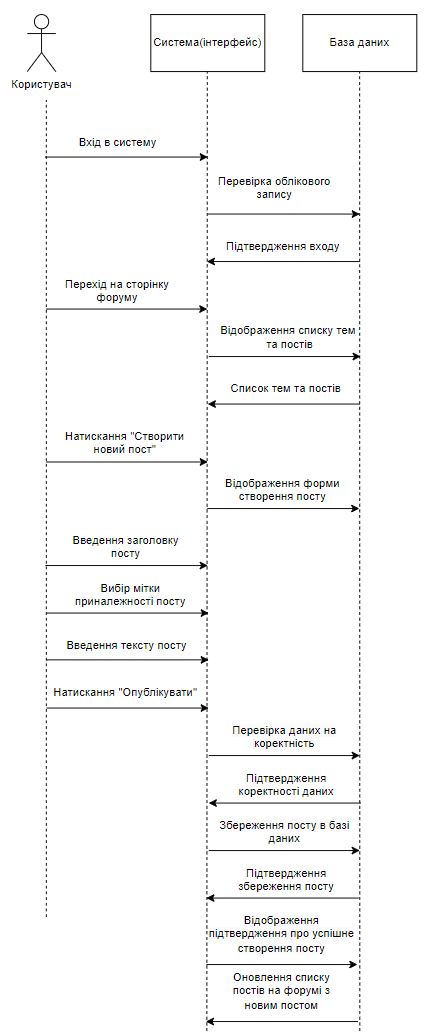


**Актори:**

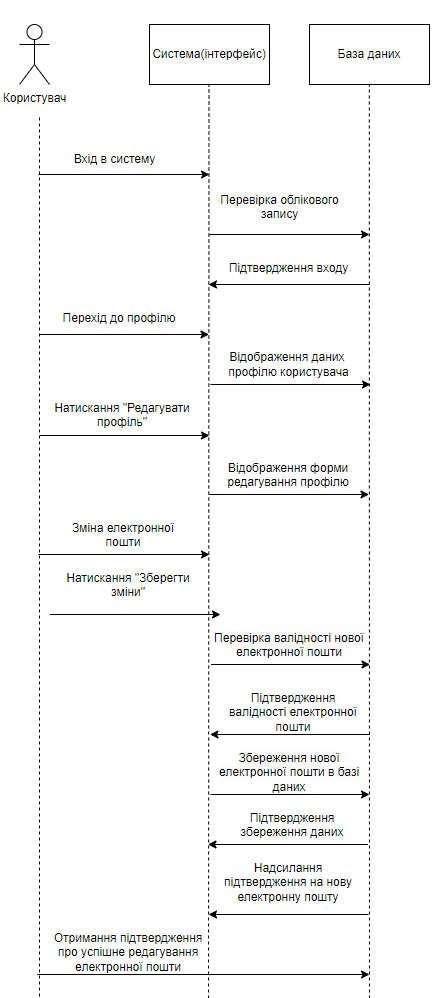
* Користувач
* Адміністратор

**Діаграма варіантів використання**



**Створення посту на форумі** 

**Редагування електронної пошти**



**Створення заходу**

