ЗМІСТ

[1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ 2](#_Toc470471634)

[2 USE-CASE ДІАГРАМИ ПРОЦЕСІВ КОРИСТУВАЧІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ 4](#_Toc470471635)

[3 ПОБУДОВА ДІАГРАМ ПОСЛІДОВНОСТЕЙ 5](#_Toc470471636)

[4 ДІАГРАМИ ІЄРХАХІЇ ПРОЦЕСІВ, ДІАГРАМИ НОТАЦІЇ СТАНУ ОБ’ЄКТІВ ДЛЯ ОСНОВНИХ ОБ’ЄКТІВ СИСТЕМИ ТА ДІАГРАМИ ПОТОКІВ ПРОЦЕСІВ 6](#_Toc470471637)

[5 ОПИС МОДЕЛЕЙ ЖИТТЄВОГО ЦИКЛУ ДЛЯ ПРОЦЕСІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ 7](#_Toc470471638)

[6 DFD 8](#_Toc470471639)

[7 ERD 10](#_Toc470471640)

# 1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ

В системі визначено наступних користувачів:

а) адміністратор ІС;

б) неавторизований користувач;

в) авторизований користувач;

г) адміністратор даних.

Користувачів кожного виду може бути більше ніж один, тобто маємо 3 базові групи користувачів.

Адміністратор ІС працює з налаштуваннями бекапу данних, оновлення даних з зовнішніх інтерфейсів, збереження та відправлення файлів на сервер.

Неавторизований користувач має наступні можливості:

- перегляд описової інформації про організовані харчування, доступні в системі;

- реєстрація/авторизація в системі.

Всі авторизовані в якості загальної області видимості даних мають власний акаунт. Користувачі в якості областей видимості даних мають:

- список організованих харчувань;

- всі види їжі з яких можна організувати харчування;

- перегляд попередніх своїх харчувань;

- вибір грошової категорії харчувань;

- вибір типу харчувань(Похудання, накачування, здорове, економне) ;

-вибір групи, курсу;

- надсилати адміністраторові повідомлень;

Адміністратор може надіслати повідомлення авторизованому користувачу або забанити його за порушення правил поведінки в системі.

При виконанні етапу курсової роботи отримано досвід аналізу користувачів та ролей у інформаційній системі на прикладі інформаційної системи «Організація харчування студентів».

# 2 USE-CASE ДІАГРАМИ ПРОЦЕСІВ КОРИСТУВАЧІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Розроблені при виконанні даного етапу курсової роботи Use Case UML діаграми складаються з наступних елементів:

а) актор;

б) процес;

в) перехід.

Актор – це об’єкт ІС, що виконує первні процеси і поділяється на:

- master (адміністратори);

- slave (решта).

Процес описує інтерфейс, що надає користувачу певний функціонал.

Перехід вказує перехід користувача між інтерфейсами та взаємозв’язок між інтерфейсами.

Переходи бувають:

- вкладені;

- розширені;

Вкладений перехід від процесу А до процесу Б означає, що Б є вкладеним підпроцесом процесу А.

Розширений перехід від процесу А до процесу Б означає, що процес Б розширює функціонал А.

Різниця між вкладеним і розширеним переходами полягає в тому, що для вкладених їх вершина є абстракцією, що реалізується через них. Для розширених переходів вершина існує як окремий модуль.

Діаграми, побудовані під час виконання даного завдання, для користувачів-не-адміністраторів мають обов'язкову точку входу – авторизацію. Всі інші процеси залежать від конкретного користувача.

При виконанні домашньої роботи отримано досвід побудови Use Case UML діаграм для користувачів інформаційної системи, що описує бізнес-процес «Організація харчування студентів».

# 3 ПОБУДОВА ДІАГРАМ ПОСЛІДОВНОСТЕЙ

а) Було побудовано діаграми послідовностей для авторизованих користувачів послідовностей: користувача Student (результати наведено у Додатку А).

б) Було описано порядок взаємодії між об’єктами послідовностей: діаграми послідовностей користувача Student.

# 4 ДІАГРАМИ ІЄРХАХІЇ ПРОЦЕСІВ, ДІАГРАМИ НОТАЦІЇ СТАНУ ОБ’ЄКТІВ ДЛЯ ОСНОВНИХ ОБ’ЄКТІВ СИСТЕМИ ТА ДІАГРАМИ ПОТОКІВ ПРОЦЕСІВ

1. Було побудовано діаграму ієрархії процесів для основних процесів системи (результати наведено у Додатку А).
2. Було побудовано діаграми нотації стану об’єктів для об’єктів типу User, типу Rozklaf Info та типу Food .
3. Було побудовано діаграми потоків процесів для процесів Authorization, Info View та Food View.

# 5 ОПИС МОДЕЛЕЙ ЖИТТЄВОГО ЦИКЛУ ДЛЯ ПРОЦЕСІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Було описано для кожного процесу інформаційної системи модель життєвого циклу, за якою його має бути розроблено. Процеси, що оперують даними користувача вимагають виского рівня безпеки, тому мають розроблюватися за задачною моделлю життєвого циклу. Процеси, для яких може виникнути потреба в швидкій зміні, мають розроблюватися за спіральною моделлю життєвого циклу.

# 6 DFD

* Розроблені в домашній роботі діаграми DFD складаються з наступних елементів:
* а) зовнішні накопичувачі (джерела інформації);
* б) процеси обробки інформації;
* в) внутрішні накопичувачі (джерела інформації) – використовуються для збереження інформації, яка отримується з процесів обробки інформації;
* г) потік даних – стрілка, яка з’єднує між собою накопичувач і процес обробки, або навпаки.
* Процеси між собою обмінюються інформацією лише через внутрішній накопичувач.
* Внутрішні накопичувачі необов’язково відображаються на діаграмах DFD нульового рівня. На ньому обов’язково зображуються зовнішні накопичувачі, головний процес обробки інформації та потоки від накопичувачів до процесу.
* Побудова діаграм DFD більш високих рівнів полягає в декомпозиції кожного процесу попереднього рівня на складові підпроцеси, що виконують більш елементарні задачі. При цьому, вже на таких діаграмах вже повинні бути внутрішні накопичувачі.
* Декомпозицію зроблено за допомогою побудови дерева процесів. При цьому всі зовнішні, внутрішні накопичувачі, що були пов’язані з основним процесом, переносяться на даний рівень діаграми DFD. На кожному наступному рівні на деяких діаграмах з’являються нові внутрішні накопичувачі.
* В рамках виконання етапу курсової роботи отримано досвід розбиття бізнес-процесу на складові підпроцеси на прикладі процесу «Організація Круїзів».

# 7 ERD

Було побудовано ERD діаграму для процесу додання та зміни інформації про конференцію, яка містить наступні таблиці:

* User\_Info – містить дані користувача, має наступні поля:

1. User\_Name – ім’я користувача;
2. User\_Surname – прізвище користувача;
3. User\_E-mail – електронна адреса користувача, що є ключом сутності;
4. User\_Password – пароль користувача;

* Food – містить наявні записи про конференції, має наступні поля:

1. Food \_Title – назва харчування;
2. Food \_Type – тип харчування;
3. Food \_DateTime – час прийому їжі;
4. Food \_kind– вид харчування студента;

Сутність Product:

1. Eat\_number;
2. Date\_of\_eat;
3. Product\_name;

Для цієї сутності визначений внутрішній ключ Eat\_number.

Для сутності Food\_details:

1. Food\_items\_id;
2. Price;
3. Food \_id;
4. Eat\_number \_fk.

Зовнішній ключ Food \_items\_id, а внутрішній ключ Food \_id \_fk, Eat\_number\_fk. Результати побудови ERD представлено у Додатку А (рис. А.1).