Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Кафедра прикладної математики

ЗВІТ

ПРО ВИКОНАННЯ VІ ЕТАПУ КУРСОВОЇ РОБОТИ

з дисципліни “Бази даних та інформаційні системи”

на тему: Організація харчування студентів

Студента ІV курсу, групи КМ-33 напряму підготовки 6.040301 – прикладна математика

Масюк О.Є.

Викладач

ТЕРЕЩЕНКО І.О.

Оцінка: \_\_\_ балів

Київ – 2016

ЗМІСТ

[ВСТУП 3](#_Toc462692885)

[1 ОПИС ЗАВДАННЯ 4](#_Toc462692886)

[2 ОПИС РЕЗУЛЬТАТІВ 5](#_Toc462692887)

[ВИСНОВКИ 14](#_Toc462692888)

[ПЕРЕЛІК ПОСИЛАНЬ 15](#_Toc462692889)

[Додаток А 16](#_Toc462692890)

[Додаток Б 18](#_Toc462692891)

# ВСТУП

Аналіз бізнес-процесу організаційної системи є важливим для побудови коректної інформаційної системи. Одним із способів такого аналізу є розбиття бізнес-процесу на складові та побудова дерева процесів на основі отриманих складових. Цей спосіб допомагає визначити які процеси треба реалізувати в інформаційній системі.

Інформаційна система – це комплекс інформаційних технологій, що реалізують бізнес процеси. Однією з технологій, що використовується для моделювання бізнес-процесів, є побудова діаграм DFD, які є графічним представленням потоків даних між процесами бізнес-процесу.

В рамках виконання даного етапу курсової роботи потрібно провести аналіз бізнес-процесу «Здача заліку» шляхом побудови діаграм DFD0, DFD1, DFD2.

# 1 ОПИС ЗАВДАННЯ

Завдання - описати бізнес-процес «Здача заліку», а саме, побудувати:

1) DFD0;

2) DFD1;

3) DFD2.

# 2 ОПИС РЕЗУЛЬТАТІВ

Виходячи з завдання, було побудовано:

1) DFD0 (додаток А);

2) DFD1 (2 підпроцеси для DFD0, додаток А);

3) DFD2 (додаток Б).

Потоки Read означають, що склад внутрішнього накопичувача повністю зчитується в процес, до якого направлена стрілочка з відповідною назвою.

Розглянемо діаграми DFD1, DFD2.

Розглянемо спочатку процес авторизації Auth та його підпроцеси (рисунки А.1 та Б.1).

Нові об’єкти (інформація про них) зберігаються на наступних внутрішніх накопичувачах:

а) Student – інформація про студентів:

1) student\_login – тип string, логін користувача-студента, ключ сутності;

2) student\_password – тип string, пароль користувача-студента;

3) student\_name – тип string, ПІБ студента;

4) student\_ng – тип string, номер групи студента, кафедри, курсу ;

5) student\_status - тип string, статус студента.

б) food info – інформація про харчування:

1) food\_id – тип string, ідентифікатор , ключ сутності (створюється системою при створенні нової дисципліни);

2) food \_name – тип string, назва харчування, ключ сутності;

3) food\_dscrptn\_info – тип text, опис харчування;

4) sbjct\_status – тип string, статус харчування.

5) food\_type- тип string, тип харчування(багатий, сердньо статистичний малобюджетний).

г) rozklad – інформація про роботи з дисципліни:

1) roz\_id – тип string, ідентифікатор розкладу харчуваня, ключ сутності;

2) roz\_id – тип string, ідентифікатор розкладу, ключ сутності (створюється системою при створенні нового харчування);

3) roz \_info - тип text, опис розкладу;

4) roz \_status – тип string, статус розкладу.

Реєстрація користувачів відбувається наступним чином. З зовнішнього інтерфейсу User, який являє собою користувача системи, відправляється інформація, яку новий користувач вводить при реєстрації, за допомогою потоку reg info у процес Auth, підпроцес Reg.

Після того, як в рамках процесу авторизації (процес Auth, підпроцес Reg) були зареєстровані нові користувачі, ці користувачі можуть увійти в систему. З зовнішнього інтерфейсу User до процесу Auth (підпроцес Login) передається логін та пароль (login, pass). Також у вказаний процес та підпроцес завантажується інформація з внутрішніх накопичувачів Student, Admin (інформація про адміністраторів). Після здійснення спроби входу, користувачу (зовнішньому інтерфейсу) надсилається інформація про статус операції входу в систему (чи є вхід успішним, чи ні). Якщо дані, які ввів користувач, існують в системі (існують на вказаних накопичувачах), то змінюється статус відповідного користувача. Зміна статусу відбувається за допомогою надсилання на відповідний внутрішній накопичувач логіну користувача (як унікального ключа для пошуку та зміни даних відповідного користувача на внутрішньому накопичувачі) та статусу користувача (який дорівнює значенню «authorized» (авторизований)) в рамках потоку authorized\_<user>, де <user> = student або teacher, або admin відповідно.

Всі користувачі при вході в систему тимчасово реєструються. Відбувається це за допомогою надсилання із зовнішньої сутності User у процес temp\_user даних ip (ip-адреса користувача) та time (час входу в систему). Результатом процесу temp\_user (тимчасова реєстрація користувача) є наступна структура даних (структура передається потоком даних reg temp user):

- temp\_user\_login - тип string, логін тимчасового користувача (його ip-адреса);

- temp\_user\_time - тип date, час входу користувача в систему.

Структура зберігається на внутрішньому накопичувачі temp\_user. Інших елементів у структурі накопичувача немає. Ключем сутності є temp\_user\_login.

Така структура даних являє собою тимчасового користувача системи. Цей користувач може зареєструватись або залогінитись під своїм логіном та паролем, або переглянути інформацію про навчальні дисципліни. Вхід тимчасового користувача в відповідні процеси відбувається за допомогою надсилання інформації в рамках потоків logining, registering, temp\_user info, структура яких співпадає зі структурою внутрішнього накопичувача temp\_user. Процеси входу тимчасового користувача під своїми даними Login та реєстрації тимчасового користувача під своїми даними Reg описано вище. Процес перегляду інформації про навчальні дисципліни Subject description view буде описано нижче.

Користувачі можуть бути видалені з системи або їх профіль може бути відредагований у разі зміни про них інформації. Їх може видалити адміністратор системи. Опис процесу має наступний вигляд. У процес видалення (зміни інформації про) користувача з системи change/drop user надходить інформація з внутрішньої сутності Admin (що являє собою вхід адміністратора в процес, потік даних administrating). Адміністратор надсилає в рамках цього процесу запит до деканату про студентів (запит students querty).

Результатом процесу є:

- зміна статусної інформації користувачів, які повинні бути видалені з системи або інформація про яких повинна бути змінена згідно інформації з деканату (потоки типу change/drop user info, структура таких потоків – це набір структур вигляду [<user>\_login; <user>\_info; <user>\_status], де <user>\_info – всі інші поля інформації про відповідного користувача, окрім паролю, <user>\_status – статус відповідного користувача (дорівнює changed у випадку зміни інформації або dropped у випадку видалення користувача з системи));

- надсилання відповідним користувачам інформації про видалення з системи або зміни інформації (потік drop status, що направлений до зовнішньої сутності User).

Для адміністратора процес входу в систему співпадає з процесом, розглянутим вище для звичайних користувачів, а процеси реєстрації в системі або видалення з неї відрізняється від процесів, розглянутих раніше для звичайного користувача. Адміністратор створюється (видаляється) розробником системи. Умовно цей процес можна показати наступним чином. З зовнішньої сутності Developer (що представляє собою розробника системи) у процес створення (видалення) адміністратора Create/drop admin надсилається інформація про адміністратора (потік даних admin data). Водночас з внутрішнього накопичувача Admin завантажується інформація про всіх наявних адміністраторів. Після створення (видалення) адміністратора надсилаються сповіщення про створення (видалення) адміністратора (потоки create/drop admin status):

- розробнику (зовнішня сутність Developer);

- користувачу, який є адміністратором, якого створили (видалили) (зовнішня сутність User).

Також дані про нового (або видаленого) адміністратора зберігаються на внутрішній накопичувач Admin. Це робиться в рамках потоку Created/droped admin, який має наступну структури:

- admin\_login – тип string, логін користувача-адміністратора, ключ сутності (внутрішнього накопичувача Admin);

- admin\_password – тип string, пароль користувача-адміністратора;

- admin\_status - тип string, статус адмінітратора, у випадку створення дорівнює created, у випадку видалення дорівнює dropped.

Внутрішній накопичувач Admin має структуру, аналогічну даному потоку даних.

Процес Auth з підпроцесами повністю розглянуто. Розглянемо тепер процес Food view (рисунок А.1 та рисунок Б.2).

Користувач може створити нове харчувння. Умовно це можна описати наступним чином. користувач входить в процес створення нового харчування New. При створенні нового харчування студент вводить необхідну інформацію про харчування.

Результатом процесу є:

- створення нового харчування (зберігається новий об’єкт в накопичувач food info за допомогою потоку даних info of created food, структура якого співпадає зі структурою відповідного йому внутрішнього накопичувача);

- створення харчування(створення множини нових об’єктів на внутрішньому накопичувачі rozklad);

Студент може передивлятись, створювати та видаляти розклади, преглядати розклади харчуваня. Все це можна робити в рамках процесу перегляду робіт Rozklad view.

Для перегляду робіт, створення нових та видалення старих розкладів. Також для цього завантажується інформація про харчування (внутрішній накопичувач food info, потік food info), інформація про роботи (внутрішній накопичувач Rozklad).

Внутрішній накопичувач EAT має наступну структуру:

- eat\_id – тип string, ідентифікатор оцінки, ключ сутності (створюється системою при створенні нової оцінки);

- student\_login – ключ сутності;

- food\_id – ключ сутності;

- roz\_id – ключ сутності;

- time\_value – тип time, час прийому;

- eat\_status – тип string, статус оцінки.

Cтуденти можуть переглядати час прийому їжі. Для того, щоб студент переглянув час їжі, в процес перегляду їжі Eat view передаються дані про прийом відповідного внутрішнього накопичувача, а також передаються дані з накопичувача food info, Eat.

Результатом процесу. Результатом є зміна статусу відповідної переглянутою прийому їжі

Всі користувачі (в тому числі і тимчасові) можуть переглядати інформацію про харчуваня. Для цього, з внутрішнього накопичувача, який відповідає користувачу, який переглядає інформацію, надсилається у процес інформація про цього користувача. Також надсилається час перегляду цієї інформації (з зовнішньої сутності User), а також інформація про харчування з внутрішнього накопичувача Food info, і всі переглянуті до цього. Результатом процесу є структура наступного вигляду (таку структуру має внутрішній накопичувач Reviewed pages): user\_login, food\_id, time.

Користувач може видаляти харчування. Для цього, в процес видалення дисципліни Del food передається інформація про rjhbcnedfxf з внутрішнього накопичувача Stutent, інформація про дисципліну з food info, інформація про оцінки з Eat, інформація про роботи з rozklad..

# ВИСНОВКИ

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».

# ПЕРЕЛІК ПОСИЛАНЬ

1. Хомоненко А.Д., Цыганков В.М., Мальцев М.Г. Базы данных: Учебник для высших учебных заведений/Под ред. проф. А.Д. Хомоненко. – СПб.: КОРОНА принт, 2002. – 672с.

2. В.В. Корнеев, А.Ф. Гареев, С.В. Васютин, В.В. Райх Базы данных. Интеллектуальная обработка информации. – М.: Нолидж, 2001.- 496с.

3. Хансен Г., Хансен Д. Базы данных. Разработка и управление. – М.: Бином, 2000. – 704 с.

4. Тимчасове положення про організацію освітнього процесу в НТУУ «КПІ» [Текст] / Уклад.: В. П. Головенкін (розд.: 1-8, 10, 12), С. В. Мельниченко (розд.: 9, 11); за заг. ред. Ю.І. Якименка. – К.: НТУУ «КПІ», 2015. – 102 с.

# Додаток А

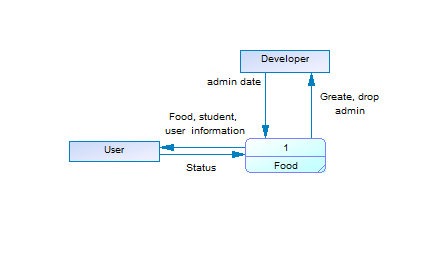


Рисунок А.1. Діаграма DFD0 (DFD нульового рівня) процесу «Організація харчування студентів»

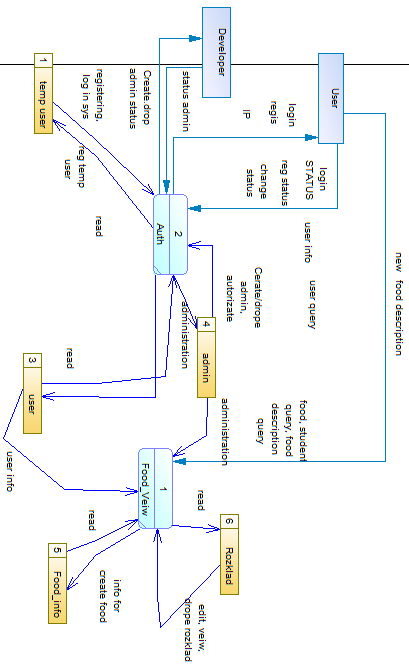


Рисунок А.2. Діаграма DFD1 (DFD першого рівня) процесу «Організація харчування студентів»

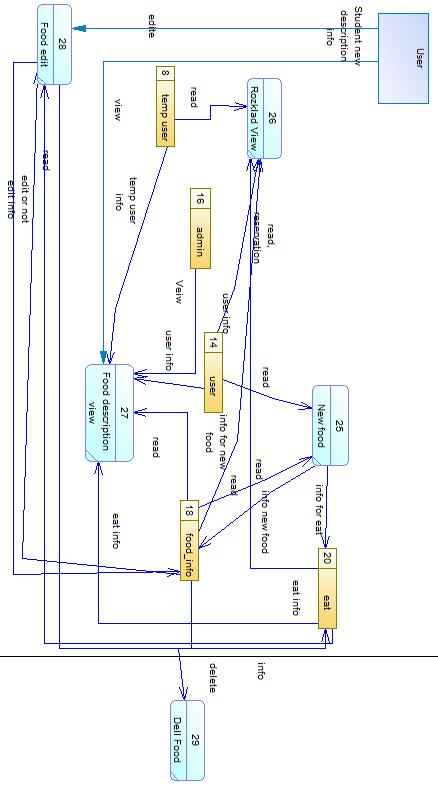


Рисунок Б.1. Діаграма DFD2 (DFD другого рівня) процесу «Організація харчування студентів», процес 1