ЗМІСТ

[1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ 2](#_Toc470450044)

[2 USE CASE UML ДІАГРАМИ 5](#_Toc470450045)

[3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ 6](#_Toc470450046)

[4 IDEF3 11](#_Toc470450047)

[5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ 14](#_Toc470450048)

[6 DFD 15](#_Toc470450049)

[7 ERD 16](#_Toc470450050)

[8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС 17](#_Toc470450051)

# 1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ

Результатом першого етапу є визначення основних груп користувачів ІС. А саме:

* неавторизований користувач;
* авторизований користувач;
* адміністратор ІС;
* адміністратор даних.

Кожна група користувачів має власну область видимості та може виконувати лише дії, що передбачені системою. Адміністратор даних працює тільки з об’єктами ІС. Адміністратор ІС може змінювати області видимості всіх інших користувачів та працює з налаштуваннями ІС. Неавторизованому користувачу доступні можливості реєстрацій або авторизації в системі.

Група авторизованих користувачів поділяється ще на одну групу користувачів: на викладачів (тобто екзаменаторів) та студентів.

Викладач може додати в систему завдання, відредагувати його, переглянути відповідь студента та поставити оцінку.

Студент може переглянути завдання, відповісти на нього та переглянути свою оцінку.

# 2 USE CASE UML ДІАГРАМИ

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).

Кожна з вищенаведених категорій побудовано UML-діаграми, які наглядно демонструють область видимості користувача та дії, передбачені системою, які він може виконувати.

Кожен з користувачів спочатку повинен або зареєструватись, або авторизуватись.

Студент та викладач мають спільні можливості: редагування профілю, перегляд інформації з предметів та, безпосередньо, екзамену, можливість листування. Але користувач «Викладач» може редагувати завдання для іспиту для свого предмету, перевіряти відповіді та ставити бали.

Натомість користувач «Студент» не має можливості редагувати завдання і оцінку, але може надсилати відповідь.

Адміністратор даних в ІС «Складання іспиту» призначає ролі користувачам, відповідає за наповнення бази даних та область видимості для користувачів «Студент» і «Викладач».

# 3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ

Діаграма послідовностей для неавторизованого користувача має такі послідовності:

Перша послідовність показує взаємодію об’єктів системи при самостійній реєстрації неавторизованого користувача. Користувач передає в процес Registration свій логін, пароль, електронну пошту, ПІБ та роль. Процес посилає ці дані до Admin для верифікації ПІБ та ролі та посилає повідомлення користувачу на електронну пошту для її підтвердження. Після того як користувач підтвердив адрес електронної пошти, перейшовши за посиланням, яке вказано в листі, інформація про активацію переходить в процес Registration, який цю інформацію передає до Admin. Admin отримує інформацію з DataServer для верифікації нового профілю. Після користувачу передається статусна інформація успішності або не успішності верифікації його профілю. У разі успішності дані профілю зберігаються на DataServer.

Друга послідовність показує взаємодію об’єктів системи при вході неавторизованого користувача під конкретними даними. Користувач передає в процес login свої логін та пароль. В процесі відбувається верифікація цих даних.

Надалі будемо вважати, що всі процеси, які потребують завантаження або відправлення даних (таких, як листи, навчальні матеріали, оцінки розклади іспитів) завантажуються або вивантажуються відповідно з DataServer.

Діаграма послідовностей для «Студента» та «Викладача» має такі послідовності: спочатку «Викладач» додає завдання для екзамену, потім «Студент» його вирішує та зберігає відповідь на сервері. Далі «Викладач» перевіряє відповіді та виставляє оцінку, яку потім повідомляє «Студенту».

Діаграма послідовностей AdminIS має такі паралельні послідовності. За допомогою першої послідовності він встановлює налаштування бекапу даних(процес Backup\_setting, на вхід процесу поступають налаштування бекапу даних). За допомогою другої послідовності він встановлює налаштування вмикання та вимикання серверу (Serve\_setting, на вхід процесу поступають налаштування вмикання та вимикання серверу). За допомогою третьої послідовності він встановлює налаштування оновлення даних (Update, на вхід процесу поступають налаштування оновлення даних).

# 4 IDEF3

На цьому етапі курсової роботи були розроблені наступні діаграми:

а) діаграма нотації всіх об’єктів системи, що беруть участь в основних процесах системи;

Для даної ІС були розглянуті наступні об’єкти: User, Mark, Task.

Об’єкт User має такі стани: not authorization, registration, login, drop.

Об’єкт Mark має такі стани: not created, new, edit, drop.

Об’єкт Task має такі стани: not created, new, edit, drop.

б) діаграма потоків основних процесів ІС.

Діаграма нотації станів об’єктів системи дозволяє зрозуміти всі стани об’єктів після певних дій.

В даному етапі для ІС «Складання іспиту» було розглянуто 3 процеси: авторизація, перегляд інформації та листування.

# 5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ

Процеси Reg, Login, Drop user, Edit profile розроблюються за каскадною моделлю життєвого циклу, бо повинні мати високий рівень безпеки, бо включають в себе роботу з (аутентифікаційними) даними користувача. Процеси New subject, New task, Edit task, Delete task, Solve task розроблюються за каскадною моделлю життєвого циклу, бо повинні мати високу якість розробки, оскільки є основотворними процесами в системі.

Всі інші процеси розробляються за спіральною моделлю, бо не мають вимог високої безпеки або якості, але в ході експлуатації можуть мати потребу в швидкій зміні або редагуванні.

# 6 DFD

При виконанні домашньої роботи отримано досвід розбиття процесу на складові підпроцеси на прикладі процесу «Складання іспиту».

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. А саме:

* DFD0 — деталізація 0-го рівня (один процес 1) представлено на рисунку А.1;
* DFD1 – деталізація першого рівня ( 3 підпроцеси 1.1, 1.2, 1.3 для процесу 1) представлено на рисунку Б.1;
* DFD2 – деталізація другого рівня ( підпроцеси 1.1.1, 1.1.2, 1.2.1, 1.2.2, 1.2.3, 1.3.1, 1.3.2) для визначених 3 підпроцесів 1.1, 1.2, 1.3, 1.4 для процесу 1) представлено на рисунку В.1, В.2, В.3.

Діаграми деталізацій рівнів DFD0, DFD1 та DFD2 побудовані за допомогою спеціального програмного забезпечення PowerDesigner.

# 7 ERD

В ході виконання даного етапу курсової роботи була проаналізована інформаційна система «Складання іспиту», для якої була розроблена ER-діаграма. Ми виділили 4 основні сутності («Student», «Teacher», «Subject», «Test») та 2 проміжні («Studtest», «Sublist»), які відображають дані про студента, викладача, предмет, завдання, їх виконання та перевірку, кінцевий результат роботи та оцінку за виконану роботу. Також були виділені внутрішні та зовнішні ключі та зазначені зв’язки між сутностями. Все це допомогло зрозуміти структуру ІС.

# 8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС

Для обох блоків процесів інформаційної системи «Складання іспиту» визначено архітектуру системи. Для блоку процесів «Авторизація користувача» в якості архітектури системи визначено «розподілене представлення даних» тому, що для даних операцій на стороні клієнта необхідна лише валідація даних, введених користувачем, а перевірка та/або збереження даних відбувається вже на сервері.

Для блоку процесів «Перегляд контенту» в якості архітектури системи визначено «розподілений застосунок» тому, що користувач може змінювати графічне представлення інформації та формувати список дисциплін за допомогою фільтрів, а також – користувач потребує можливості пакетно відправляти дані (методичні матеріали).