ЗМІСТ

[1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ 2](#_Toc470467517)

[2 USE CASE UML ДІАГРАМИ 4](#_Toc470467518)

[3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ 5](#_Toc470467519)

[4 IDEF3 7](#_Toc470467520)

[5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ 8](#_Toc470467521)

[6 DFD 9](#_Toc470467522)

[7 ERD 10](#_Toc470467523)

[8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС 11](#_Toc470467524)

# 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 2 базові групи користувачів.

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

Неавторизований користувач має наступні можливості:

- реєстрація/авторизація в системі.

Вся множина авторизованих користувачів поділяється на дві групи, тобто в системі, окрім базових груп користувачів (ролей), існує ще дві групи (ролі):

а) викладач;

б) студент.

Ці дві групи в якості загальної області видимості даних мають власний акаунт. Викладачі в якості областей видимості даних мають:

- списки груп, в яких вони викладають з усіма оцінками за заняття або виконання робіт, пов’язаних лише з цим викладачем;

- перелік своїх колоквіумів;

- здані цьому викладачу або перевірені цим викладачем роботи;

- повідомлення, надіслані студентам, або від студентів;

Області видимості студента є наступними:

- список доступних дисциплін;

- список колоквіумів з даних дисциплін;

- свої здані/перевірені роботи;

- списки своїх оцінок;

- повідомлення від викладачів/викладачам.

Адміністратор даних підтверджує реєстрацію в системі викладачів та студентів. Підтвердження відбувається також на основі отриманих з зовнішнього інтерфейсу «Деканат» даних. Також 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 «Здача заліку».

# 2 USE CASE UML ДІАГРАМИ

Результатом виконання даного етапу являються готові use case діаграми для усіх типів користувачів. З них видно, що неавторизований користувач може лише зареєструватися або зайти в систему, студент може подивитися на завдання та відповісти на них, а також дізнатися свою оцінку, а викладач може створювати нові завдання та оцінювати успішність студентів.

# 3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ

Для користувача Student було виділено наступні процеси:

1. Authorization;
2. Validation;
3. Messaging;
4. Work with tasks.

При вході в систему користувач Student надсилає запит процесу Authorization з параметрами Login і Password, які потім передаються процесу Validation, який звіряє їх із наявними у базі даних даними. В разі успішної звірки даних, процес Validation надсилає підтвердження процесу Authorization, який повідомляє про успіх користувача Student.

У рамках роботи з процесом Messaging користувач Student може виконувати наступні дії: надсилання, читання та видалення повідомлень.

У рамках роботи з процесом Work with tasks користувач має спочатку вибрати певне завдання, а потім відправити свою відповідь.

Для користувача Lector було виділено наступні процеси

1. Authorization;
2. Validation;
3. Messaging;
4. Work with colloquiums;
5. Work with students.

При вході в систему користувач Lector надсилає запит процесу Authorization з параметрами Login і Password, які потім передаються процесу Validation, який звіряє їх із наявними у базі даних даними. В разі успішної звірки даних, процес Validation надсилає підтвердження процесу Authorization, який повідомляє про успіх користувача Lector.

У рамках роботи з процесом Messaging користувач Lector може виконувати наступні дії: надсилання, читання та видалення повідомлень.

У рамках роботи з процесом Work with colloquiums користувач Lector має змогу продивлятися, додавати чи видаляти колоквіуми.

У рамках роботи з процесом Work with students користувач Lector має змогу передивитися список студентів, передивитися відповідь студента та поставити за неї оцінку.

# 4 IDEF3

В рамках виконання даного етапу курсової роботи було створено ієрархічне дерево процесів, для об’єктів системи було побудовано діаграми нотації станів, а для основних процесів системи – діаграми потоків процесів. З ієрархічного дерева видно, які саме процеси будуть у ІС, а на інших типах діаграм можна побачити життєві цикли об’єктів.

# 5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ

Процеси Registration, Login, Drop\_user розроблюються за V-моделлю життєвого циклу, бо належать до процесів, пов’язаних із особистими даними користувачів, тому мають забезпечувати достатній рівень безпеки. Процеси Creating\_colloquium, Editing\_colloquium, Answering\_colloquium, Putting\_marks розроблюються за V-моделлю життєвого циклу, бо являються основними процесами системи і було б доцільно забезпечити найвищу якість їх розробки

Всі інші процеси розробляються за задачною моделлю, бо не відображають дуже важливі функції, а задачна модель дозволяє спростити розробку.

# 6 DFD

На DFD0 знаходиться основний процес Writing colloquium та три зовнішні сутності: Student, Lecturer, Admin.

На DFD1 основний процес декомпонується на три наступні процеси: Work with users, Work with tasks, Work with marks. Процес Work with marks працює із зовнішньою сутністю Lecturer, яка перевіряє свою роль, а тоді надсилає оцінку у внутрішній накопичувач Answers.

На DFD2 процесу Work with users цей процес декомпонується на три наступні процеси: Login, Registration, Drop. Усі зовнішні сутності передають процесу Registration свої логіни, паролі та ролі, процесу Login – лише логіни та паролі, а процесу Drop зовнішня сутність Admin передає логін користувача, якого необхідно видалити. Усі процеси працюють з єдиним внутрішнім накопичувачем users.

На DFD2 процесу Work with tasks цей процес декомпонується на чотири наступні процеси: Creating, Editing, Deleting, Answering. З першими трьома працює зовнішня сутність Lecturer, відправляючи їм інформацію про завдання, яке слід створити, редагувати або видалити, попередньо перевіривши свою роль у внутрішньому накопичувачі users. Lecturer у цьому процесі працює із внутрішнім накопичувачем tasks. Student працює з процесом Answering, попередньо перевіривши свою роль та дізнавшись завдання із внутрішнього накопичувача tasks. Тоді Student надсилає свою відповідь у внутрішній накопичувач Answers.

# 7 ERD

Під час виконання даного етапу було створено ER-діаграму.

Таблиці Student та Lecturer мають наступні поля: Login, Password, Email, Name, Surname. Усі вони мають тип string(20). Таблиця Task має наступні поля: Author, Question, TaskName, CorrectAnswer. Усі вони мають тип string(20). Таблиця User\_Answer має наступні поля: Author\_fk, Lecturer\_fk, TaskName\_fk, Answer, Mark, ColloquiumName\_fk. Усі вони, крім Mark, який має тип int, мають тип string(20). Тяблиця Colloquium має два поля типу String(20): ColloquiumName та Author\_fk.

# 8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС

Під час виконання даного етапу було обрано архітектури для блоків UseCase діаграм. Для блоку Authorization було обрано архітектуру розподіленого представлення. Для блоків View\_Colloquium, Work\_With\_Students, Work\_With\_Answers, Edit\_Colloquium було обрано архітектуру віддаленого представлення.