ЗМІСТ

[1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ 2](#_Toc470469602)

[2 USE CASE UML ДІАГРАМИ 4](#_Toc470469603)

[3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ 5](#_Toc470469604)

[4 IDEF3 6](#_Toc470469605)

[5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ 8](#_Toc470469606)

[6 DFD 9](#_Toc470469607)

[7 ERD 11](#_Toc470469608)

[8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС 12](#_Toc470469609)

1 РОЛІ КОРИСТУВАЧІВ В ІНФОРМАЦІЙНІЙ СИСТЕМІ

Були визначені наступні групи користувачів: неавторизований користувач, студент, викладач, адміністратор даних на рівні факультету, адміністратор даних на рівні інституту та адміністратор ІС..

Неавторизованому користувачу доступні наступні операції:

1. перегляд основної інформації про систему, що дозволяє зрозуміти чи потрапив користувач на потрібний ресурс;
2. реєстрація нового користувача – студента;
3. вхід до системи по логіну та паролю.

Студенту надається доступ до наступних операцій:

1. перегляд всіх призначених для нього завдань;
2. виконання та здача призначених для нього завдань до відповідного часу, після настання якого здати роботу неможливо;
3. перегляд результатів своїх робіт;
4. перегляд списків студентів своєї навчальної групи;
5. перегляд/редагування свого облікового запису.

Викладачу надається доступ до наступних операцій:

1. перегляд/редагування/створення/видалення своїх завдань;
2. призначення завдання конкретному студенту або групі, що закріплені за даним викладачем;
3. перегляд списків студентів всіх груп призначених до даного викладача;
4. виконувати перевірку та виставляти оцінку для тих виконаних(зданих) завдань, які були призначені даним викладачем;
5. переглядати всі результати виконаних завдань, які були перевірені саме даним викладачем;
6. перегляд/редагування свого облікового запису.

Адміністратору даних на рівні факультету відкритий доступ до наступних операцій:

1. реєстрація(створення)/видалення облікових записів викладачів;
2. управління навчальними групами(створення, редагування);
3. призначення навчальних груп до викладачів;
4. підтвердження реєстрації студента(перевірка навчальної групи);
5. генерація звітності(успішність студентів по групам, факультету; статистика по викладачам і т.п.);
6. перегляд/редагування свого облікового запису.

Адміністратор даних на рівні інституту створюється розробником та має наступні права:

1. управління факультетами(створення, редагування);
2. реєстрація(створення)/видалення облікових записів викладачів;
3. призначення викладача адміністратором даних на рівні факультету;
4. генерація звітності;
5. перегляд/редагування свого облікового запису.

Адміністратор ІС відповідає за налаштування та підтримання працездатності системи вцілому. Адміністратор ІС створюється розробником.

2 USE CASE UML ДІАГРАМИ

За допомогою програмного забезпечення Power Designer були створені use case діаграми для студента, викладача, адміністратора на рівні факультету, адміністратора на рівні інституту та для неавторизованого користувача.

Студент має за основні функції: написання контрольної роботи.

Викладач відповідно створює завдання та надає до виконання студенту.

Головний адміністратор створює та видаляє адміністраторів на рівні інституту,а також має доступ до створення та редагування навчальних груп, викладачів, студентів, управління зв’язками між ними.

# 3 ДІАГРАМИ ПОСЛІДОВНОСТЕЙ

Під час виконання роботи були створені діаграми послідовностей для студента та викладача.

Як зареєстрований користувач, студент спочатку має ввійти в систему за допомогою свого логіна та пароля. Після цього студент може перейти до написання тесту, перегляду свого профілю або до перегляду результатів своїх робіт.

Для написання тесту студент має спочатку перейти до сторінки перегляду тесту, вказуючи унікальний номер тесту. Після отримання тесту користувач може розпочати його виконання, після виконання якого йде відправка результатів. Студент отримує сповіщення, що сигналізує, або про вдалу відправку, або ж про помилку, що сталася в момент відправки.

Для управління своїм обліковим записом студент повинен перейти на сторінку профіля, де можна отримати всю доступну інформацію, а також, за бажанням, перейти до сторінки редагування облікового запису.

Для перегляду отриманих балів за тест, користувач повинен перейти до сторінки результатів, де йому надається повна інформація по виконаній роботі.

Після авторизації в інформаційній системі викладач може перейти до перевірки написаних тестів, до управління тестами, або до управління обліковим записом.

Для перевірки тестів викладач обирає роботу, що хоче відкрити та після цього переходить на сторінку перегляду результатів тесту. Тут користувач отримує роботу студента та може перейти безпосередньо до перевірки відповідей. Після перевірки викладач повинен виставити оцінку роботі.

# 4 IDEF3

Для процеса «написання контрольної роботи» були виділені два основних процеси: User maintain – пов'язаний з безпекою та роботою з користувачами та Work with test – процес з основною бізнес-логікою майбутньої ІС. Управління користувачами у свою чергу розбивається на авторизацію(реєтрація та логін) та роботу з обліковим записом(перегляд та редагування). Робота з тестом поділяється на такі процеси: управління (створення, перегляд, редагування, видалення), написання (процес написання та відправки), перевірка та закріплення тестів за студентами.

Розглянемо об’єкт «Test». Спочатку контрольна робота створюється, потім для того щоб робота мала певний сенс її потрібно наповнити. Далі, після наповнення, роботу можна присвоїти певному студенту або ж, при необхідності, знищити. Викладач може забрати роботу у студента, якщо робота була помилково призначена не тому студенту. Коли студенту присвоїли роботу він може розпочати виконання контрольної роботи і призупинити виконання в певних випадках. Завершивши роботу, студент відправляє свої відповіді для перевірки викладачу. Після цього викладач може перевірити роботу. Також викладач може заново перевірити роботу у разі помилки під час оцінювання або у разі скарги студента. Перевірена або нікому не присвоєна контрольна робота через певний період часу може бути знищена.

Об’єкт «студент» може перебувати у чотирьох статусах:

а) Новий студент – статус, що отримує обліковий запис після подання запиту на реєстрацію.

б) Зареєстрований студент – статус, що отримує обліковий запис після підтвердження введених даних після реєстрації. Облікові записи з даним статусом можна використовувати для входу у систему.

в) Авторизований студент – статус, що отримує обліковий запис, коли користувач ввійшов в ІС використовуючи даний аккаунт.

г) Видалений студент – статус, що отримує обліковий запис після видалення. Запис може бути видалений на етапі підтвердження у разі введення користувачем невірних даних та видалений у разі довготривалої неактивності.

На основі вище наведених міркувань були побудовані діаграми нотації станів та process flow(IDEF3) діаграми для розглянутих об’єктів.

# 5 МОДЕЛІ ЖИТТЄВИХ ЦИКЛІВ РОЗРОБКИ ОСНОВНИХ ПРОЦЕСІВ СИСТЕМИ

На відміну від інших процесів, для процесу авторизації обрана каскадна модель життєвого циклу, тому що не допускається впровадження «сирої» системи, адже йте роботе з даними користувача, втрата яких може принести значні збитки.

Для інших процесів краще підходить спіральна модель, тому що за допомогою обраної моделі в короткий проміжок часу можна отримати частково готовий продукт з основною функціональністю, що вже буде приносити прибуток та можна вносити корективи до системи.

Безпосередньо сам процес розробки курсової роботи більш схожий на спіральну модель, так як за кожен чітко виділений період повинна бути представлена певна вже готова функціональність і після консультації з викладачем визначаються нові вимоги, що повинні бути реалізовані. І так повторюється поки не будуть виконана вся запланована функціональність.

# 6 DFD

Під час виконання етапу були створені Data Flow діаграми нульового, першого та другого рівнів.

У діаграмі нульового рівня зображене просте уявлення про створювану інформаційну систему вцілому: студент виконує контрольну роботу; викладач створює завдання та перевіряє виконані роботи; а адміністрація, у свою чергу, створює план контрольних робіт.

У діаграмі першого рівня головний процес розбивається на основні підпроцеси системи, що дозволяє дослідити потік даних в системі. У нашому випадку можна побачити, що студент надає свої дані системі, що зберігаються, для подальшого використання в процесі обробки контрольної роботи. Аналогічно, дані про викладача зберігаються в системі і використовуються під час створення завдань та перевірки контрольної роботи. З діаграми також видно, що адміністрація надає системі інформацію про графіки контрольних та групи студентів. Графіки контрольних робіт використовуються для ініціалізації журналів.

Далі, проаналізуємо процес обробки контрольної роботи. Як бачимо до цього надходять дані про студента, викладача, завдання та журнал, що відображає хід та результати проходження контрольної роботи. Також процес оновлює дані журналу(наприклад, початок та кінець контрольної роботи), додає результати студента в журнал(створюючи лінію журналу) та зберігає виконану роботу. Під час перевірки потрібний також доступ до написаних котрольних робіт.

Під час створення Data Flow діаграми другого рівня були розбиті процеси авторизації студента(реєстрація та логін), обробка завдань(створення, редагування та видалення), обробка журналу(ініціалізація та підтвердження журналу для подальшого опублікування), процес контрольної роботи(написання та перевірка) та обробка даних адміністрації(створення груп та графіку написання контрольних робіт).

# 7 ERD

Під час виконання етапу були виділені ключові сутності та зв’язки, що між ними встановлюються для процесу «Написання контрольної роботи» і на основі цього була створена ERD. У якості ключових сутностей були взяті:

1. User – користувач системи;
2. Student;
3. Teacher;
4. Group – група студентів;
5. Group\_teacher – зв'язок між викладачами та групами, що відображає з якими групами викладач працює;
6. Exercise – окреме завдання;
7. WorkTemplate – заготовка завдань роботи, що може бути видана студентам;
8. Exercise\_WorkTemplate – зв’язок між заготовкою роботи та окремими завданнями, що вказує на список завдань в роботі;
9. StudentWork – робота студента(відповідь), що може бути оцінена;
10. Ledger – журнал, що містить інформацію про контрольну роботу: викладач, час та результати.

Створену ERD можна використовувати при проектуванні баз данних.

# 8 АРХІТЕКТУРИ СИСТЕМИ ДЛЯ ОСНОВНИХ ПРОЦЕСІВ ІС

Під час виконання етапу було обрано та пояснено архітектуру для кожного блоку Use case діаграми.

Проаналізувавши результати можна побачити, що використовувалися лише дві архітектури: розподілене представлення даних та віддалене представлення даних. Причому розподілене представлення даних значно переважає.

Тобто можна дійти висновку, що то вся логіка роботи з даними для майбутнього застосунку будуть відбуватись на стороні сервера, а на стороні клієнта буде реалізоване введення та виведення даних з відповідним форматуванням, а також можливість зберігання деяких даних без з’єднання до мережі.

Це дозволяє при зміні логіки обробки інформації не оновлювати всіх клієнтів, тобто зміна логіки буде відбуватись централізовано.

Також слід відмітити, що при даній архітектурі складність розробки серверної частини буде значно складніша, ніж розробка клієнтської частини.