**Практична робота №6. Узгодження діаграм прецедентів, класів та послідовностей.**

**Мета: Навчитися аналізувати предметну область при розробці діаграм прецедентів, класів та послідовностей засобами мови UML та застосувати їх на практиці.**

**Завдання:**

1. Опрацювати теоретичні відомості. Перевірити засвоєння матеріалу на контрольних запитаннях.
2. Доповнити табл.1 прецедентами, які з Вашої точки зору не враховані.
3. Доповнити розроблені на ПР№2 діаграми класів відповідно до розглянутого матеріалу в розділі "Теоретичні відомості та методичні вказівки".
4. Доповнити розроблену в ЛР№1 діаграму випадків використання / прецедентів з урахуванням вимог в доповненій Вами табл.1 , а також з погляду пропозицій з Рис.2. Звертаємо увагу, що на Рис.2 представлені не всі вимоги, тобто визначена не вся функціональність надана в табл.1.
5. Доповнити розроблену в ЛР№1 діаграму випадків використання / прецедентів власними прецедентами, які б включали відношення включення та розширення.
6. До створеного на ЛР№3 проекту додати діаграми послідовностей щодо реалізації прецедентів "Фіксація успішності" та "Отримання відомостей про склад груп студентів по навчальним предметам".
7. Оформити звіт, до якого включити табл.1, розроблені діаграми та Ваші письмові пояснення щодо зроблених Вами удосконалень.
8. Надіслати на електронну адресу викладача та/або викласти на платформу коледжу у власний репозитарій, під’єднаний до репозитарію викладача, звіт та модель (файл xmi), створену в Umbrello UML Modeller. Ім’я xmi-файлу задайте таке ж як і звіту.
9. Оформити звіт, до якого включити розроблені діаграми. Файл надавати з іменем у форматі

**ОPI<Номер групи><Номер лекції / практичної / лабораторної>[-<Номер завдання>][літера позначення типу роботи L – лекція, P –практична, R – лабораторна]<Прізвище англійською>**. Наприклад, **ОPI3104L**buts.doc. Відповіді на запитання повинні бути не довгими і змістовними. Не копіюйте фрагментів з різних інформаційних джерел, подумайте і викладіть свою точку зору. При наявності відповідей-"близнюків" відповідь буде зараховуватися першому за часом надсилання.

**Строк виконання цієї роботи ІПЗ-31 25.02.2021, ІПЗ-32 25.02.2021**

## Теоретичні відомості та методичні вказівки.

Побудова діаграм прецедентів, класів та послідовностей при розробці програмного забезпечення (ПЗ) є важливим етапом щодо визначення та узгодження вимог замовника, аналізу предметної області, оцінювання розробником поставлених завдань, визначення необхідних інформаційних ресурсів, потрібної функціональності ПЗ, конкретизації завдань, які повинно вирішувати ПЗ, та послідовності реалізації цих завдань. Це питання, які лежать на поверхі, а при створенні діаграм будуть виявлятися ще нові виклики, які потрібно буде вирішувати. Цю роботу потрібно проводити до написання першого рядка коду. Зазвичай це серія обговорень команди розробника для обрання стратегії розробки (ЖЦ ПЗ, методологія розробки) та розроблення стратегічного плану з конкретизацією та етапністю щодо реалізації замовленої функціональності ПЗ.

На ЛР№2 було поставлено завдання: для гіпотетичного навчального закладу потрібно створити застосунок, який би дозволяв викладачам визначати склад груп, яким викладаються їхні предмети, проставляти підсумкову оцінку, студентам – визначати предмети, які вони повинні вивчати, викладача цього предмету, а адміністрації навчального закладу отримувати списки груп та інформацію про успішність. Це завдання ми будемо вирішувати протягом наших ЛР та ПР, спираючись на теоретичний матеріал лекцій, ЛР та ПР.

Першим кроком при розробці ПЗ є визначення вимог. Для цього потрібно визначити користувачів як акторів з певними ролями. Акторами згідно з постановкою задач у нас є Адміністратор, Викладач, Студент. Звертаю увагу, що це ролі, які "виконують" конкретні користувачі. Адміністратором може бути як окрема призначена для цього особа, так в викладач, якому це доручили, або студент, який поєднує навчання з роботою в цьому навчальному закладі і який це виконує за посадовими обов’язками. Тобто по суті через акторів визначається основна функціональність ПЗ.

Можна представити на початковому етапі зв’язок акторів такою діаграмою класів (рис.1).

 Рисунок 1. Діаграма класів на початковому етапі.

Для визначення вимог в ЛР№2 потрібно було заповнити таблицю з описом вимог. Прецедент це по суті функція, яку повинне надавати ПЗ конкретному користувачу, якого визначають як актора з роллю.

Таблиця з описом вимог може бути, наприклад, у вигляді табд.1.

Таблиця 1. Описом вимог

|  |  |
| --- | --- |
| Прецедент | Дійова особа - актор |
| Введення та коригування даних про викладачів, студентів та, при потребі, інших осіб | Адміністратор |
| Формування звітності | Адміністратор |
| Фіксація успішності | Адміністратор, Викладач |
| Отримання відомостей про склад груп студентів по навчальним предметам | Адміністратор, Викладач, Студент |
| Отримання відомостей про склад навчальних предметів групи студентів | Адміністратор, Студент |

З таблиці вимог можна побачити, що інформації наданої на рис.1 в діаграмі класів недостатньо для вирішення завдань, поставлених у вимогах. Потрібен перелік груп, перелік предметів і встановлені зв’язки з класами "Викладач" і "Студент".

Перейдемо до розгляду прецедентів. Об’єднаємо всі обов’язки Адміністратора по обробленню даних в сценарій "Ведення інформаційної бази", який при уточненні виявиться значно складнішим ніж надане в табл.1 представлення. Наприклад, якщо почнемо розглядати дії, пов’язані з виявленням помилок у вхідних даних. Діаграма прецедентів у спрощеному вигляді надана на Рис.2. Списки і звітність попередньо повинна бути сформована, і це відноситься до обов’язків Адміністратора. І Викладач, і Студент повинні мати можливість або самостійно отримати потрібні їм списки, або через запит до Адміністратора. Тут одразу визначається необхідність для створення окремих інтерфейсів для Адміністратора та Викладача разом зі Студентом. Це повинні бути інтерфейси з різною функціональністю, оскільки Адміністратор може ще виконувати коригування (наприклад, вносити зміни у заголовки списків).

Кожен прецедент в процесі обговорення буде деталізований та уточнений. Наприклад, після формування звіту про успішність може бути сформований список відмінників чи список на відрахування; або при зверненні до ПЗ потрібно ввести автентифікацію, відповідно до якої визначаються права доступу до тих чи інших функцій.

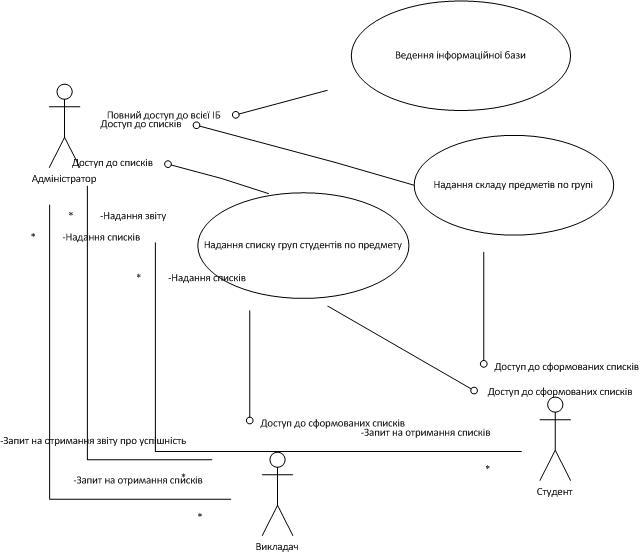


Рис.2 Діаграма прецедентів

Тепер, коли у нас є загальний погляд на функціональність ПЗ, можна визначати послідовність дій щодо її реалізації.

Головними елементами діаграм послідовності є ***об’єкти***, які є логічними сутностями, що представляють окремі елементи системи та ***повідомлення***, якими вони обмінюються. В якості об’єктів можуть виступати також актори. Повідомлення можуть бути не тільки абстрактними діями, що виконуються, але і методи класів, створених на діаграмі класів. Повідомлення на діаграмі послідовності пронумеровані, тобто мають чітку послідовність.

Наприклад, реалізацію прецеденту "Отримання відомостей про склад навчальних предметів групи студентів" можна представити як послідовність дій на Рис.3.

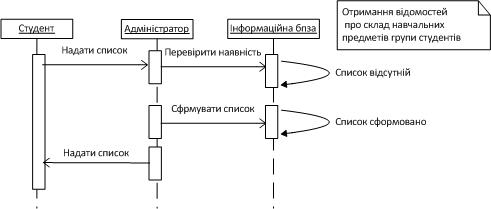


Рис.3. Отримання відомостей про склад навчальних предметів групи студентів

**Контрольні запитання**.

1. Який основний вид діаграм у концептуальній моделі
2. Яке призначення логічної моделі?
3. Назвіть основний вид діаграм у фізичній моделі.
4. Що описують діаграми випадків використання ?
5. Що являє собою варіант використання?
6. Які зв’язки можуть мати випадки використання з іншими випадками використання?
7. Що на діаграмі випадків використання визначає актор?
8. Які стандартні види відносин існують між акторами й варіантами використання?
9. На які групи можна поділити UML-діаграми?
10. Яка стратегія використання *UML*-діаграм при моделюванні програмних систем?
11. Яке призначення діаграм використання?
12. Яке призначення діаграм класів?
13. Якими відносинами можуть бути пов'язані сутності діаграми?
14. Які типи візуальних позначень існують в UML?
15. Які діаграми відносяться до статичних?
16. Чи може статична діаграма бути і поведінковою?
17. Що являє собою діаграма класів?
18. Яке призначення діаграми класів?
19. Дайте визначення поняттю «клас».
20. Поясніть базові відношення в UML. Що являє собою асоціація? У чому зміст множинності асоціацій? У чому відмінність атрибутів від асоціацій?
21. Що являє собою операція класу? У чому зміст узагальнення?
22. До якої групи діаграм відноситься діаграма послідовності?
23. Що описує діаграма послідовності ?
24. Як вказується плин часу на діаграмі послідовності
25. Як зображується лінія життя об'єкта на діаграмі послідовності?
26. Що таке фокус керування на діаграмі послідовності?
27. Як зображується фокус керування на діаграмі послідовності?
28. Чи може об'єкт посилати повідомлення самому собі?
29. Як на схемі різняться рефлексивне повідомлення та рекурсія
30. Як можна визначити повідомлення на діаграмі послідовності?
31. Які види повідомлень на діаграмі послідовності ви знаєте? (асинхронний виклик, виклик процедури, повернення з процедури, загублені та знайдені повідомлення)