

УДК 378.1

Гриценко Валерій Григорович, кандидат педагогічних наук, доцент, завідувач кафедри автоматизації та комп'ютерно-інтегрованих технологій Черкаського національного університету імені Богдана Хмельницького, м. Черкаси, e-mail: grycenko@ukr.net

Луценко Галина Василівна, кандидат фізико-математичних наук, доцент, доцент кафедри автоматизації та комп'ютерно-інтегрованих технологій Черкаського національного університету імені Богдана Хмельницького, м. Черкаси

UML-МОДЕЛЮВАННЯ ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ “НАВЧАЛЬНИЙ ПЛАН”

Анотація

Актуальність матеріалу, викладеного у статті, обумовлена потребами сучасного університету у створенні інформаційно-аналітичних систем управління навчальним процесом. Розглядаються особливості проектування автоматизованої інформаційно-аналітичної системи управління навчальним навантаженням засобами середовища Rational Rose. Використовуючи мову UML, побудовано основні діаграми моделі інформаційно-аналітичної системи. Проведено аналіз предметної галузі, визначено функції і завдання автоматизованої інформаційної системи, що проектується.

Розроблені діаграми формують основу шаблону структури і зв'язків між елементами системи і дозволяють відслідковувати різні аспекти її поведінки. У подальшому, розроблена система може бути доповнена діаграмами для генерації коду.

Ключові слова: навчальний план, навчальне навантаження, інформаційно-аналітична система, мова UML.

Вступ. За сучасних умов, ефективна робота з інформацією є невід'ємним атрибутом кожної потужної, конкурентоспроможної організації. Формування єдиного інформаційного простору, що охоплюватиме всі аспекти організаційно-педагогічного забезпечення цілісності процесу управління в навчальному закладі, є комплексною задачею. Відповідно, для забезпечення належного рівня гнучкості роботи з базовою інформацією і швидкого реагування на її зміни, кожен навчальний заклад повинен

впроваджувати ефективні сучасні технології. Важливим критерієм, що допомагає оцінити рівень ефективності впровадження інформаційно-комунікаційних технологій, є можливість постійно розвиватися, отримувати й аналізувати інформацію, адаптуватися до зовнішніх умов.

Останнім часом у багатьох ВНЗ України почали впроваджувати локалізовані засоби автоматизації окремих напрямів управлінської діяльності (системи обліку документів і кореспонденції, системи обліку кадрів, системи управління бібліотеками, системи бухгалтерського обліку тощо). Але неможливість синхронізації дій між різними системами, відмінності у представленні інформації, і як наслідок, складність передавання даних між ними, породжують нові проблеми, пов'язані з надлишковою інформацією, неоперативністю отримання потрібних даних, фрагментарністю виконання єдиних процесів. Усе це зводить нанівець намагання суб'єктів освітньої діяльності перекласти її рутинні процеси на засоби інформаційно-комунікаційних технологій.

Методи та засоби дослідження. Найбільш вдалим рішенням у цій ситуації є використання сучасних мережевих технологій і єдиного сховища даних, що надає можливість реалізувати простий механізм інтеграції інформації в єдиний інформаційний ресурс вищого навчального закладу і забезпечити можливість сумісного використання інформації з урахуванням механізму розмежування доступу (з метою захисту даних) багатьма користувачами: адміністрацією ВНЗ, викладачами, студентами різних форм навчання, абітурієнтами. Це надає можливість досягти високого рівня цілісності даних і створюються умови для реалізації комплексної автоматизованої системи управління навчальним процесом університету.

Під час створення такої системи слід виділити, принаймні, дві проблеми з якими неминуче стикаються розробники. По-перше, і це головне, створювані системи є складними і дуже складними системами і проектами. І не важливо, переважає в проекті “технічна складність” (статична) або “динамічна складність управління”. Складність програмних систем зростає в міру створення нових версій. І в якийсь момент настає “ефект критичної маси”, коли подальший розвиток ІС стає неможливим, оскільки вже ніхто не представляє в цілому “що і чому відбувається”. Відбувається втрата управління проектом. Наступною проблемою є організація змістовного спілкування між замовниками і розробниками. Жарт про те, що

“замовник щось хоче, але точно не знає, що саме”, часто виявляється реальністю. А якщо на початковому етапі роботи над проектом ІС замовник думає, що точно знає, що хоче, то, як правило, і про це свідчить досвід авторів, його вимоги змінюються в ході виконання проекту. З одного боку, апетит приходить під час їжі, а з іншого боку, висока динаміка процесів у сучасній освіті об’єктивно змушує змінювати вимоги до ІС, що розробляється.

Способом вирішення визначених вище проблем є використання в аналізі, проектуванні й розробці інформаційних системи об’єктно-орієнтованих технологій (ООТ). Такі технології, ґрунтуючись на тісному взаємозв’язку процесів і даних у системах, дозволяють програмним системам бути надійнішими, легшими для реалізації і стійкішими до змін. Крім того, така філософія моделювання найбільше відповідає загальним концепціям поведінки систем реального світу [1, 2, 5].

З появою ООТ виникла потреба в інструментальних засобах, які дозволяють отримувати набагато складніший порівняно із звичайною блок-схемою опис процесів. Базовими поняттями, якими оперують системи з ООТ, є об’єкти і класи об’єктів. З іншого боку, можливість моделювати складні системи і процеси, що в них протікають, поєднується з інтуїтивно зрозумілими способами подання. Це дозволяє людям, які не мають технічної підготовки брати активну участь в проектуванні.

На думку авторів, найвдалішим способом подолання вищезазначених проблем є використання мови UML (Unified Modeling Language – UML). Вибраний розробниками UML (які створювали мову моделювання, а не мову програмування), підхід дозволяє архітекторам систем ефективно описувати класи, методи і зв’язки між ними [4, 7].

Перевагами обраної предметної галузі була можливість консультуватися і, відповідно, враховувати побажання безпосередніх учасників навчального процесу до інформаційної системи, яка моделювалася. Отримані неформальні вимоги були перетворені в артефакти мови UML.

Постановка задачі. Базовою задачею, що постає перед розробниками, є реалізація комплексної автоматизованої системи управління навчальним процесом університету з такими функціями:

1. Розробка (модернізація) навчальних планів за напрямами і спеціальностями і робочих навчальних планів на наступний навчальний рік.

2. Розрахунок штатів професорсько-викладацького складу університету.
3. Розподіл навчального навантаження викладачів кафедр.
4. Формування розкладу занять навчальних груп в університеті.
5. Облік проміжної (рейтинг) і підсумкової успішності студентів.
6. Формування додатку до диплома.

Із зазначених функцій нами було досліджено, спроектовано й реалізовано в АІСУ “Навчальне навантаження”, що складається з підсистем:

- “Навчальний план”;
- “Робочий навчальний план”;
- “Розрахунок навчального навантаження”;
- “Навчальне навантаження кафедри”;
- “Навчальне навантаження викладача”.

Навчальний план є базовим документом, що визначає зміст професійної підготовки фахівця. У ньому реалізуються основні принципи відбору предметів, їх систематизація, регламентується обсяг навчальних дисциплін, навантаження студента за періодами навчання, форми підсумкового контролю знань. На структуру і зміст навчального плану впливають ОКХ і ОПП за відповідними напрямками і спеціальностями, інструктивні матеріали МОН України [6], вимоги підприємств і організацій – безпосередніх замовників фахівців, тощо.

Підсистема “Навчальний план” (рис. 1) відповідає за розробку (модернізацію) навчальних планів за напрямками, спеціальностями і спеціалізаціями. Головними складовими підсистеми є: блок генерації графіка навчального процесу, блок формування множини навчальних дисциплін і блок визначення форм і видів державної атестації.

Процес формування навчального плану складається з етапів:

- створення заголовка;
- створення графіка навчального процесу;
- створення переліку дисциплін;
- створення переліку видів і форм державної атестації.

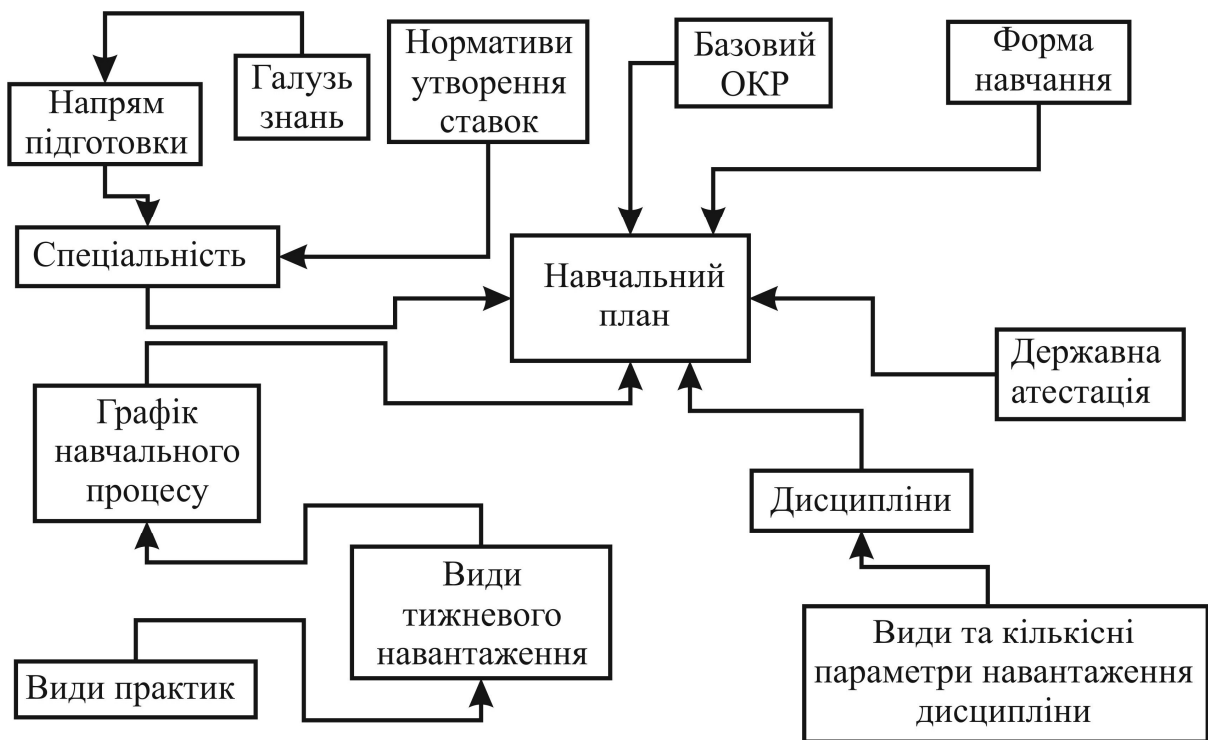


Рис. 1. Структура навчального плану

Створення заголовка навчального плану передбачає введення низку параметрів. Ці параметри поділяються на дві групи інформативні й інформативно-дієві. До інформативних відносимо: базовий освітньо-кваліфікаційний рівень (актуальний для підготовки магістрів, чи спеціалістів після бакалаврату), кваліфікацію і спеціалізацію. До інформаційно-дієвих відносимо: форму навчання (впливатиме на процес формування графіка навчального процесу), нормативи практик тощо, термін навчання (визначає кількість семестрів, і також впливає на графік навчального процесу), підпорядкування (визначає належність навчальному підрозділу і права доступу до інформації), спеціальність (надає інформацію про напрям підготовки, галузь знань тощо).

Наступним етапом є формування графіка навчального процесу, який здійснюється автоматично, з урахуванням раніше визначених параметрів форми і терміну навчання і завчасно підготовлених шаблонів графіка навчального процесу. Елементами шаблону графіка навчального процесу є види тижневого навантаження для 52 тижнів календарного року. На процес використання того чи іншого шаблону впливають: форма і термін навчання, напрям підготовки, підпорядкування та інші чинники.

У випадку, коли тижневе навантаження є певним видом практики, тоді графік навчального процесу стає повноцінним носієм інформації про практику, і саме ця інформація використовується для подальших розрахунків навантаження.

Задля спрощення процесів формування похідних від навчального плану документів, у підсистему заздалегідь закладається додаткова, неактуальна у поточний момент, інформація. Так, зокрема, у процесі формування переліку спеціальностей, додається нормативний параметр кількості студентів для утворення однієї ставки викладача, який буде використано під час розподілу кількості ставок між формами і елементами навчального навантаження.

На етапі формування переліку навчальних дисциплін, окрім основних складових, зокрема, загальної кількості годин, кількості лекційних, лабораторних і практичних годин тощо, вводиться кілька додаткових параметрів: статус дисципліни, об'єднання дисциплін у потік та ін.

Усі додаткові параметри банку дисциплін навчального плану, крім статусу дисципліни, є неактуальними у процесі формування навчального плану, але знадобляться у процесі автоматичного формування робочого навчального плану і під час оптимізації навчального навантаження викладачів.

Статус дисципліни є важливим і визначальним параметром для чіткого розмежування дисциплін на державну складову і варіативну складову, це також дає можливість вносити до навчального плану необмежену кількість альтернативних дисциплін (за вибором навчального закладу, за вибором студента).

На етапі формування банку дисциплін закладаються основи оптимального використання робочого часу викладачів за рахунок об'єднання у потоки однакових або споріднених дисциплін з різних навчальних планів, ще на стадії їх формування.

Заключним етапом формування навчального плану є створення переліку видів і форм державної атестації. Окрім основних параметрів: форма атестації і семестр, банк форм державної атестації містить основні параметри, а саме, форму атестації і семестр і додатковий параметр – статус, який, як і переважна більшість інших додаткових параметрів, що існують в системі, знадобиться для подальшої обробки даних. Зокрема, для банку державної атестації параметр статус розмежовує складові державної атестації на екзамен і захист кваліфікаційної роботи (проекту), що стане в пригоді під час формування навантаження викладача й індивідуального плану

навчання студента.

Отримані результати. Побудова діаграми прецедентів

У процесі розв'язання поставленої задачі було виконано розробку низки базових діаграм мови UML. Поведінка системи описується за допомогою функціональної моделі, яка відображає системні прецеденти, системне оточення (діючі особи або актори) і зв'язки між прецедентами й акторами (діаграми прецедентів). Основна задача моделі прецедентів – бути єдиним засобом, що дає можливість замовнику, кінцевому користувачеві й розробнику разом обговорювати функціональність і поведінку системи.

Розробка моделі прецедентів розпочинається на стадії задумки з вибору акторів і визначення загальних принципів функціонування системи. Потім на етапі опрацювання модель доповнюється детальною інформацією до існуючих прецедентів, а в разі потреби додаються нові [1].

Тому, розробку діаграми Варіантів використання розпочинаємо з визначення Діючих осіб (Actor) нашої предметної галузі, намагаючись якнайповніше врахувати всі розгалуження і варіанти поведінки (табл. 1).

Таблиця 1

Опис діючих осіб та їх функцій

Діюча особа	Короткий опис
Методист	Працівник університету (директор інституту, декан, завідувач кафедри), який для кожного з підрозділів університету формує навчальні плани дисциплін і має можливість як вводити інформацію, так і редагувати чи видаляти її. Також, методист формує робочі навчальні плани
Користувач	Працівник університету (викладач), який може переглянути таблиці з інформацією про відповідні навчальні дисципліни (кількість годин, форми контролю і т. д.), здійснювати друк цих даних, але не має можливості їх редагувати
Системний адміністратор	Визначає права доступу. Слідкує за виглядом, наповненням і режимом функціонування відповідної ІС. Підтримує існуючі й розробляє нові шаблони введення/виведення даних. Виконує автоматичний розрахунок навчального навантаження для кафедр і викладачів за попередньо введеними даними

На рис. 2 подано головну діаграму прецедентів для нашої системи. Спільною операцією для всіх акторів є авторизація.

Наступним кроком під час побудови UML моделі є формування Потoku подій – послідовності подій, потрібних для забезпечення необхідної поведінки. Даний потік описується в термінах того, “що” система повинна робити, а не “як” вона повинна це робити. Наведемо приклад потоку подій для прецеденту “Формування навчального плану”. Обов’язковою умовою під час формування потоку подій є визначення передумов, за яких даний потік може мати місце, умов розгалуження й результатів виконання тих чи інших дій [2].

Потік подій для прецеденту “Формування заголовку навчального плану”

1.1. Передумови

Підпотік створення масивів з різними типами базових освітньо-кваліфікаційних рівнів, кваліфікацією, спеціалізацією, формою навчання, термінами навчання, підпорядкуванням, спеціальністю повинен бути виконаний Системою раніше. Методист використовує відповідні дані з цих масивів для формування заголовку навчального плану.

1.2. Головний потік

Прецедент починає виконуватися, коли Методист підключається до системи реєстрації і вводить свій пароль. Система перевіряє правильність пароля (E-1) і просить Методиста обрати потрібний підрозділ (E-2). Система пропонує потрібну операцію – створити новий навчальний план (Add), видалити (Delete), редагувати (Change), надрукувати (Print) чи вийти (Quit).

Якщо обрана операція додати (Add), S-1: виконується потік додати навчальний план.

Якщо обрана операція видалити (Delete), S-2: виконується потік видалити навчальний план.

Якщо обрана операція редагувати (Change), S-3: виконується потік редагувати навчальний план.

Якщо обрана операція надрукувати (Print), S-4: виконується потік надрукувати навчальний план.

Якщо обрана операція видалити (Quit), S-5: виконується потік вийти.

1.3. Підпотоки

S-1: додати навчальний план

Система відображає вікно діалогу, яке містить поля для введення підпорядкування, спеціальності, спеціалізації, форми навчання, базового ОКР, терміну навчання й кваліфікації. Методист обирає відповідні варіанти (E-3). Система створює навчальний план (E-4). Потім прецедент розпочинається спочатку.

S-2: Видалити навчальний план

Система відображає вікно діалогу, яке містить перелік навчальних планів. Методист обирає відповідний навчальний план (E-5). Система його видаляє (E-6). Потім прецедент розпочинається спочатку.

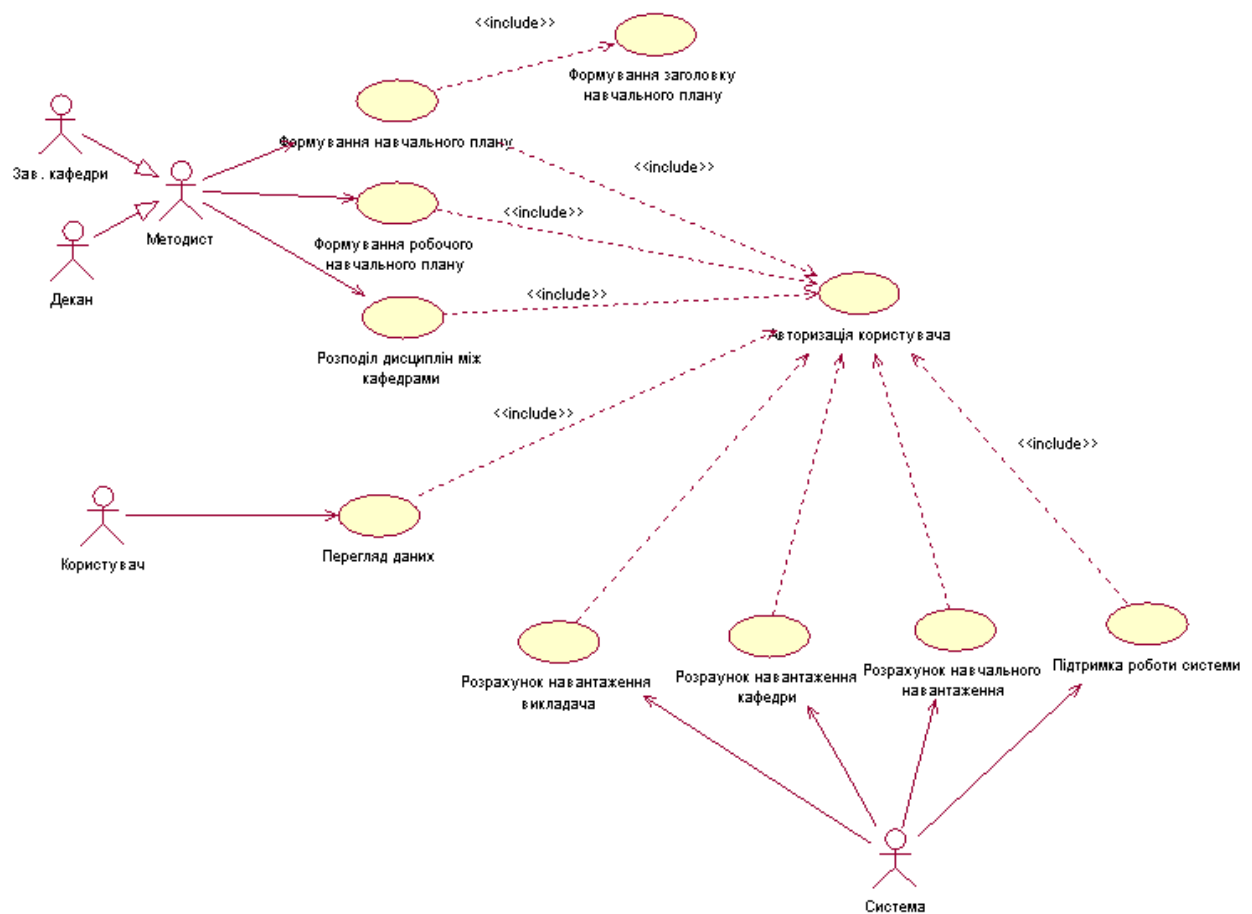


Рис. 2. Головна діаграма прецедентів

S-3: Змінити навчальний план

Система відображає вікно діалогу, яке містить поля з переліком навчальних планів. Методист обирає потрібний навчальний план. Система відкриває меню з параметрами даного навчального плану. Методист обирає потрібні характеристики (E-7). Система зберігає внесені зміни. Прецедент розпочинається спочатку.

S-4: Надрукувати навчальний план

Система отримує навчальний план і друкує його (E-8). Прецедент розпочинається спочатку.

1.4. Альтернативні потоки

E-1: введено хибний пароль. Методист повинен повторити введення чи завершити прецедент.

E-2: введено хибний підрозділ. Користувач повинен повторити введення чи завершити прецедент.

E-3: залишилися незаповнені поля чи формат даних не відповідає, передбаченому системою. Методист повинен відредагувати дані чи завершити прецедент.

E-4: заголовок навчального плану не може бути відображений. Методисту повідомляється, що дана команда на поточний момент є недоступною. Прецедент розпочинається знову.

E-5: обраний план неможливо відредагувати. Прецедент розпочинається спочатку.

E-6: система не може видалити відповідний навчальний план. Інформація зберігається, Система може видалити відповідний об'єкт пізніше. Виконання прецеденту продовжується.

E-7: залишилися незаповнені поля чи формат даних не відповідає, передбаченому системою. Методист повинен відредагувати дані чи завершити прецедент.

E-8: система не може виконати друк даних. Методисту повідомляється, що дана опція на поточний момент недоступна. Прецедент розпочинається заново.

Документи з описом потоку подій формуються і зберігаються окремо від даних програми Rational Rose, але вони пов'язані з прецедентами.

Побудова діаграми діяльності

Діаграми діяльності – відображають динаміку проекту і є схемами потоків управління в системі від дії до дії, а також паралельні дії й альтернативні потоки. Діаграми діяльності ілюструють дії, переходи між ними, елементи вибору й лінії синхронізації [1].

На рис. 3 подано діаграму діяльності для дій зі створення навчального плану.

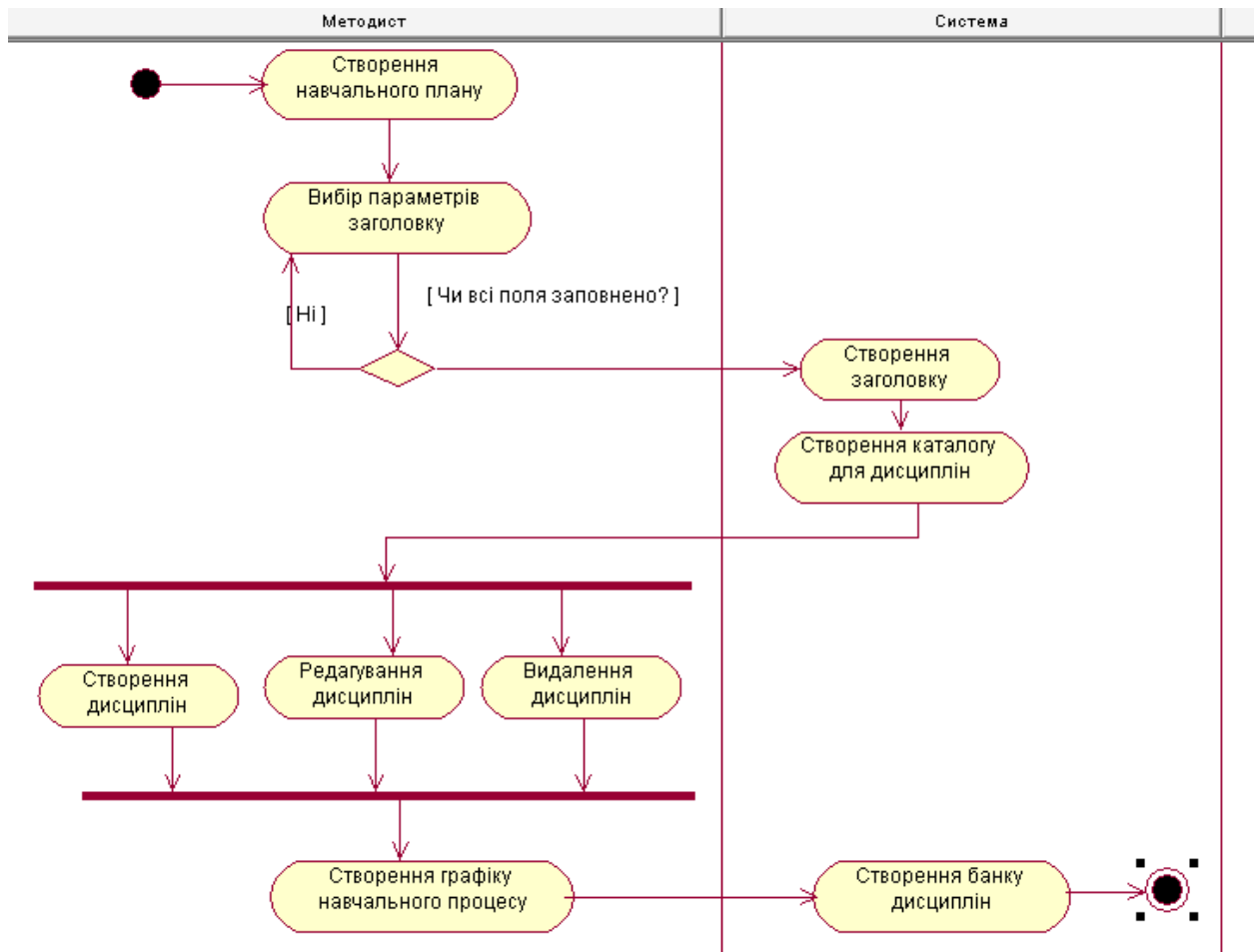


Рис. 3. Діаграма діяльності зі створення навчального плану

Дана діаграма містить дії й переходи між ними. Також на діаграмі діяльності можна вказувати умовні переходи, які визначають дії системи під час виконання чи невиконання певної умови. За допомогою секцій, що поділяють діаграму на декілька ділянок, можна встановити, хто відповідає за виконання дій на кожній з них.

Побудова діаграми послідовності

Далі визначаються основні об'єкти системи й порядок дій під час їх взаємодії. Для цього в UML-моделі використовуються діаграми Послідовності. Такі діаграми формуються для всіх блоків використання [3, 4].

Для створення діаграми послідовності розпочинаємо роботу з пакетом Logical View, вибравши пункт **Sequence Diagram**. Побудова розпочинається з розташування на ній об'єктів, які будуть обмінюватися повідомленнями. Спочатку потрібно розмістити об'єкти, які надсилають повідомлення, а потім об'єкти, що їх отримують.

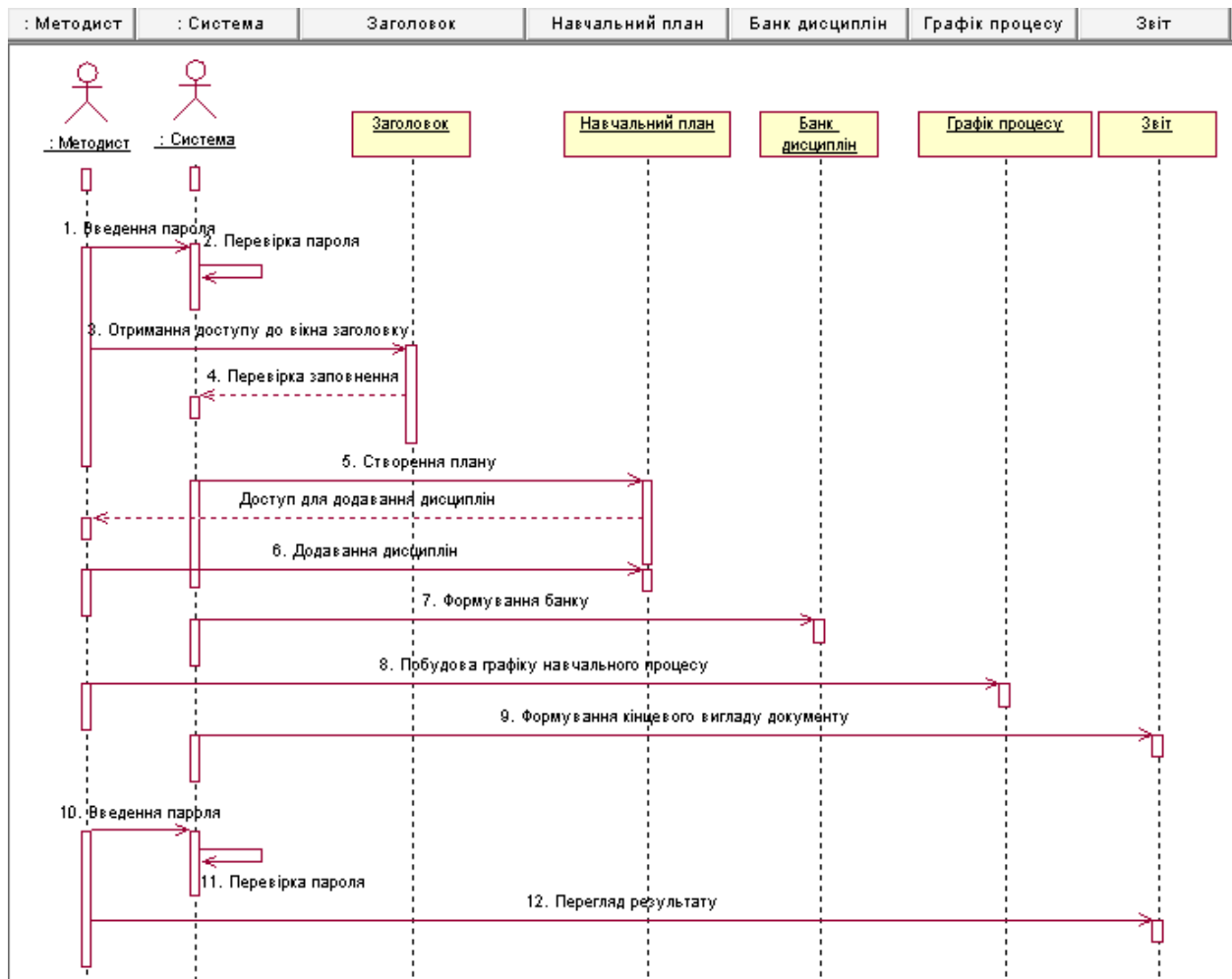


Рис. 4. Діаграма послідовності

Аналогічно формуються діаграми для всіх подій.

Висновки. Розвиток інформаційних технологій охоплює все ширші сфери людської діяльності й активними учасниками цього процесу стають працівники, що не мають освіти, пов'язаної з комп'ютерними технологіями. Використання засобів візуального моделювання, як це показано для автоматизованої інформаційної системи «Навчальне навантаження», дозволяє максимально оптимізувати процес розробки системи, узгоджуючи вимоги кінцевих користувачів продукту й вимоги, пов'язані з процесом розробки програмного продукту. Представлений проект автоматизованої інформаційної системи має прикладне значення для організації роботи професорсько-викладацького складу за сучасних умов функціонування університету.

Список використаних джерел

1. Буч Г. UML. Руководство пользователя / Буч Г., Рамбо Дж., Джекобсон А. – М. : ДМК Пресс, 2003. – 432 с.: ил.

2. *Кватрани Т.* Rational Rose 2000 и UML. Визуальное моделирование / Т. Кватрани; Пер. с англ. – М. : ДМК Пресс, 2001. – 176 с.: ил. – (Серия «Объектно-ориентированные технологии в программировании»).

3. *Куперштейн. В.* Современные информационные технологии в производстве и управлении / В. Куперштейн. – СПб. : БХВ, 2000. – 304 с.

4. *Леоненков А.* Самоучитель UML / А. Леоненков. – [2-е издание]. – СПб. : БХВ-Петербург, 2004. – 432 с.

5. *Мацяшек Л. А.* Анализ и проектирование информационных систем с помощью UML 2.0 / Л. А. Мацяшек. – М. : Вильямс, 2008. – 816 с.

6. *Болюбаш Я. Я.* Організація навчального процесу у вищих закладах освіти : навч. посібник для слухачів закладів підвищення кваліфікації системи вищої освіти / Я. Я.Болюбаш. – К. : ВВП «КОМПАС», 1997. – 64 с.

UML-МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННО-АНАЛИТИЧЕСКОЙ СИСТЕМЫ “УЧЕБНЫЙ ПЛАН”

Гриценко Валерий Григорьевич, кандидат педагогических наук, доцент, заведующий кафедрой автоматизации и компьютерно-интегрированных технологий Черкасского национального университета имени Богдана Хмельницкого, г. Черкассы, e-mail: grycenko@ukr.net

Луценко Галина Васильевна, кандидат физико-математических наук, доцент, доцент кафедры автоматизации и компьютерно-интегрированных технологий Черкасского национального университета имени Богдана Хмельницкого, г. Черкассы

Аннотация

Актуальность материала, изложенного в статье, обусловлена потребностями современного университета в создании информационно-аналитических систем управления учебным процессом. Рассматриваются особенности проектирования автоматизированной информационно-аналитической системы управления учебной нагрузкой средствами UML. Используя язык UML, построены основные диаграммы модели информационно-аналитической системы. Проведен анализ предметной области, определены функции и задачи проектируемой автоматизированной информационной системы.

Разработанные диаграммы формируют базу шаблона структуры и связей между элементами системы и позволяют отслеживать разные аспекты ее поведения. В

последующем, разработанная система может быть дополнена диаграммами для генерации кода.

Ключевые слова: учебный план, учебная нагрузка, информационно-аналитическая система, язык UML.

UML-DESIGN OF THE INFORMATION ANALYSIS SYSTEM "CURRICULUM"

Valeriy G. Gritsenko, PhD, associate professor, the Head of the Department of Automation and Computer Integrated Technologies, Cherkasy National University named after Bogdan Khmelnytsky, Cherkasy, e-mail: grycenko@ukr.net

Galina V. Lutsenko, PhD, associate professor of the Department of Automation and Computer Integrated Technologies, Cherkasy National University named after Bogdan Khmelnytsky, Cherkasy

Resume

Relevance of the material, which is presented in papers, is conditioned by the requirements of the state-of-the-art universities in creation of the information-analysis systems of the education management. The main peculiarities of the design of the automated information-analysis management system by using UML are considered. The main diagrams of the information system were created by using the UML. The knowledge domain was analyzed; the functions and tasks of the automated system were defined.

The designed diagrams organize the base of the pattern of the structure and connections between the system elements and allow to trace the different aspects of its behavior. Hereafter the designed system might be supplemented by the diagrams to code generation.

Keywords: curriculum, teaching load, information-analytical system, UML.

Матеріал надійшов до редакції 02.04.2011 р.