Лабораторна робота №1

Побудова за допомогою програми Rational Rose UML діаграми варіантів використання автоматизованої інформаційної системи

***Мета роботи:*** *засвоїти методику та виробити практичні в побудові за допомогою програми Rational Rose UML діаграми варіантів використання автоматизованої інформаційної системи.*

*Теоретичні відомості*

**Огляд Rational Rose**

Rational Rose - потужний CASE-засіб для проектування програмних систем будь-якої складності. Одним із переваг цього програмного продукту є можливість використання діаграм на мові UML. Можна сказати, що Rational Rose є графічним редактором UML діаграм.

У розпорядження проектувальна система Rational Rose надає наступні типи діаграм, послідовне створення яких дозволяє отримати повне уявлення про всю проектовану систему і про окремі її компоненти:

Use case diagram (діаграми прецедентів);

Statechart diagram (діаграми станів);

Activity diagram (діаграми активності);

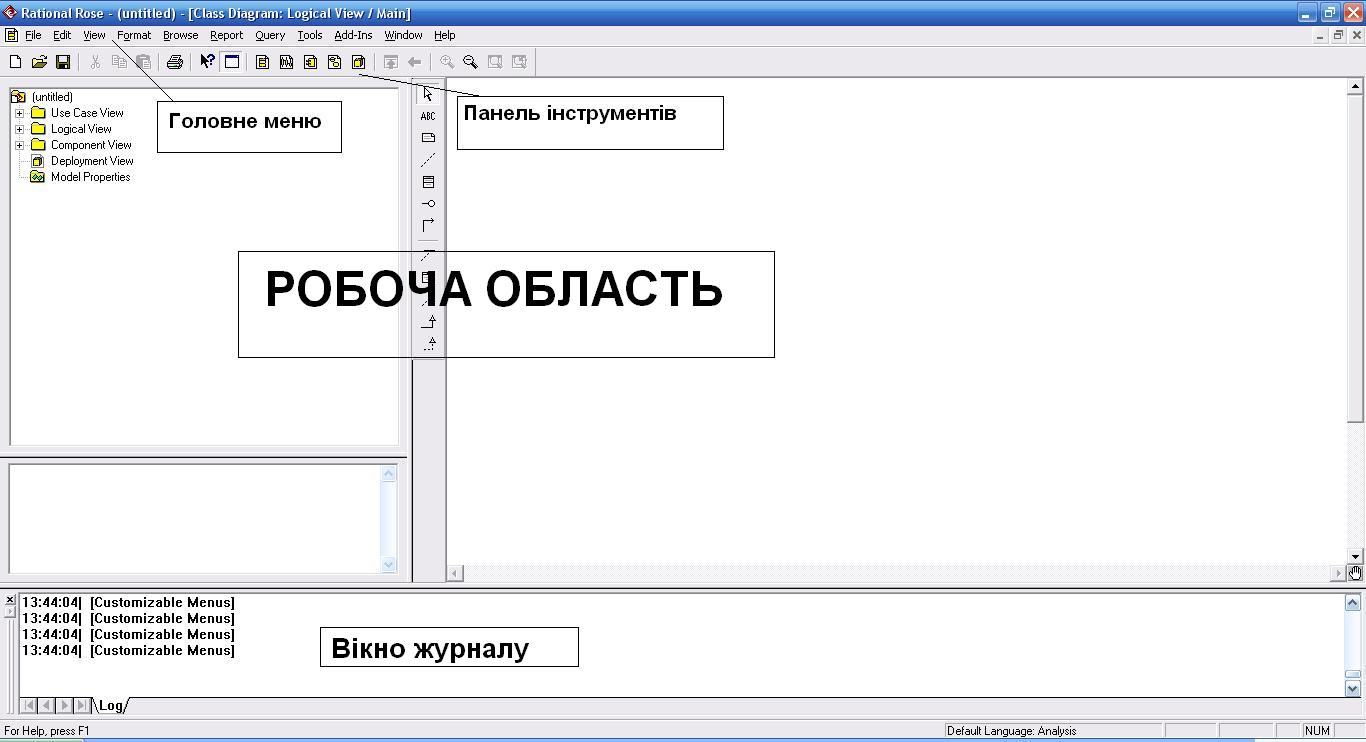
Interaction diagram (діаграми взаємодії);

Sequence diagram (діаграми послідовностей дій);

Class diagram (діаграми класів);

Component diagram (діаграми компонент).

У Rational Rose реалізовані загальноприйняті стандарти на робочий інтерфейс програми. Після встановлення Rational Rose на комп'ютер, запуск цього засобу (без вибору готових шаблонів проектів) призводить до появи на екрані відповідного робочого інтерфейсу, в якому автоматично створюється новий проект і в робочому вікні діаграми з'являється за замовчуванням (рис. 1).



Головне меню програми Rational Rose виконано в загальноприйнятому стандарті і має наступний вигляд (рис. 2).



Окремі пункти меню об'єднують подібні операції, пов'язані до всього проекту в цілому. Деякі з пунктів меню містять добре знайомі операції, такі як відкриття проекту, виведення на друк діаграм, копіювання в буфер і вставка з буфера різних елементів діаграм. Інші операції настільки специфічні, що можуть знадобитися додаткові зусилля для їх вивчення (властивості операцій генерації програмного коду або перевірки узгодженості моделей).

Стандартна панель інструментів розташовується нижче рядка головного меню і має наступний вигляд (рис. 3). Деякі з інструментів недоступні для нового проекту, який не має ніяких елементів. Стандартна панель інструментів забезпечує швидкий доступ до тих команд меню, які найбільш часто виконуються розробниками.



Користувач може налаштувати зовнішній вигляд цієї панелі на свій розсуд. Для цього необхідно виконати операцію головного меню: Tools/Options ( Інструменти/Параметри ), відкрити вкладку Toolbars (Панелі інструментів) діалогового вікна, і натиснути кнопку Standard (Стандартна) . У відкритому вікні можна переносити необхідні кнопки з лівого списку в правий список, а непотрібні кнопки - з правого списку в лівий. Даним способом можна показати або приховати різні кнопки інструментів, а також змінити їх розмір.

Операції головного меню File (Файл) дозволяють створювати нові моделі в нотації мови UML, завантажувати і зберігати розроблювану модель в зовнішньому файлі та роздруковувати на принтері розроблені діаграми .

Операції головного меню Edit (Редагування) дозволяють виконувати дії з редагування елементів моделі і їх властивостей, а також виконувати пошук елементів в рамках розроблюваного проекту.

Операції головного меню View (Вид) дозволяють відображати на екрані різні елементи робочого інтерфейсу і змінювати графічне представлення діаграм. Призначення операцій цього пункту головного меню представлено у наступній таблиці ( табл. 1) .

|  |  |  |
| --- | --- | --- |
| Назва | Зображення | Призначення |
| Toolbars |  | дозволяє налаштувати зовнішній вигляд робочого інтерфейсу системи Rational Rose і містить додаткові підпункти |
| Standard |  | робить видиму/невидиму стандартну панель інструментів |
| Toolbox |  | робить видиму/невидиму стандартну панель інструментів поточної активної діаграми |
| Configure |  | викликає діалогове вікно налаштування параметрів моделі |
| Status Bar |  | робить видимий/невидимий рядок стану |
| Documentation |  | Робить видимим/невидимим вікно документації |
| Browser |  | Робить видимим/невидимим браузер проекту. |
| Log |  | Робить видимим/невидимим вікно журналу |
| Editor |  | Робить видимим/невидимим вбудований текстовий редактор. |
| Time Stamp |  | Включає/вимикає режим відображення часу в записах журналу |
| Zoom to Selection |  | Змінює масштаб зображення виділених елементів моделі , так щоб вони розмістилися в одному вікні |
| Zoom In |  | Збільшує масштаб зображення |
| Zoom Out |  | Зменшує масштаб зображення |
| Fit in Window |  | Змінює (зменшує) масштаб зображення всіх елементів поточної діаграми, так щоб всі вони розмістилися в одному вікні |
| Undo Fit in Window |  | Скасовує зміну масштабу зображення розміщення елементів в одному вікні |
| Page Breaks |  | Розбиває поточну діаграму на сторінки для подальшого друку |
| Refresh |  | Перемальовує поточну діаграму |
| As Booch |  | Зображує елементи моделі відповідно до нотацією Г. Буча |
| As OMT |  | Зображує елементи моделі відповідно до нотацією OMT |
| As Unified |  | Зображує елементи моделі відповідно до нотацією мови UML |

Операції головного меню Format (Формат) дозволяють виконувати дії по зміні зовнішнього вигляду елементів моделі на різних діаграмах. Призначення операцій цього пункту головного меню представлено у наступній таблиці (табл. 2) .

|  |  |
| --- | --- |
| Назва | Призначення |
| Font Size | Змінює масштаб використовуваного шрифту |
| Font | Викликає діалогове вікно вибору шрифту |
| Line Color | Викликає діалогове вікно вибору кольору ліній |
| Fill Color | Викликає діалогове вікно вибору кольору для зображення графічних елементів діаграм |
| Use Fill Color | Включає/вимикає режим відображення кольору для зображення графічних елементів діаграм |
| Automatic Resize | Включає / вимикає режим автоматичного зміни розмірів графічних елементів діаграм для відображення текстової інформації про їх властивості |
| Stereotype | Дозволяє вибрати спосіб зображення стереотипів виділених елементів діаграми і містить додаткові підпункти : |
| None | стереотип не відображається |
| Label | стереотип відображається у формі тексту |
| Decoration | стереотип відображається у формі невеликої піктограми в правому верхньому куті графічного елемента |
| Icon | елемент діаграми відображається у формі спеціального графічного стереотипу , якщо даний стереотип передбачений у програмі |
| Stereotype Label | Включає / вимикає режим відображення текстових стереотипів для взаємозв'язків ( асоціацій , залежностей і пр. ) діаграми |
| Show Visibility | Включає / вимикає режим відображення кванторів видимості атрибутів і операцій виділених класів |
| Show Compartment Stereotypes | Включає / вимикає режим відображення текстових стереотипів атрибутів і операцій виділених класів |
| Show Operation Signature | Включає / вимикає режим відображення сигнатури операцій виділених класів |
| Show All Attributes | Робить видимими / невидимими атрибути виділених класів |
| Show All Operations | Робить видимими / невидимими операції виділених класів |
| Suppress Attributes | Робить видимою / невидимою секцію атрибутів виділених класів . Приховує секцію атрибутів навіть у тому випадку , коли обрана опція Show All Attributes |
| Suppress Operations | Робить видимою / невидимою секцію операцій виділених класів . Приховує секцію операцій навіть у тому випадку , коли обрана опція Show All Operations |
| Line Style | Дозволяє вибрати спосіб графічного зображення ліній взаємозв'язків і містить додаткові підпункти : |
| Rectilinear | лінія зображається у формі вертикальних і горизонтальних відрізків |
| Oblique | лінія зображається у формі похилих відрізків |
| Toggle | проміжний варіант зображення лінії |
| Layout Diagram | Дозволяє автоматично розмістити графічні елементи у вікні діаграми з мінімальною кількістю перетинів і накладень з'єднувальних ліній |
| Autosize All | Дозволяє автоматично змінити розміри графічних елементів поточної діаграми таким чином , щоб текстова інформація містилася всередині зображень відповідних елементів |
| Layout Selected Shapes | Дозволяє автоматично розмістити виділені графічні елементи у вікні діаграми з мінімальною кількістю перетинів і накладень з'єднувальних ліній |

Операції головного меню Browse (Огляд) дозволяють відображати робочі вікна з різними діаграмами і викликати діалогові вікна редагування властивостей окремих елементів моделі. Призначення операцій цього пункту головного меню представлено у наступній таблиці ( табл. 1.3).

|  |  |  |
| --- | --- | --- |
| Назва | Наявність кнопки на стандартній панелі | Призначення операції головного меню |
| Use Case Diagram |  | Викликає діалогове вікно з пропозицією вибрати для відображення в робочому вікні одну з існуючих діаграм варіантів використання моделі або приступити до розробки нової діаграми |
| Class Diagram |  | Викликає діалогове вікно з пропозицією вибрати для відображення в робочому вікні одну з існуючих діаграм класів або приступити до розробки нової діаграми |
| Component Diagram |  | Викликає діалогове вікно з пропозицією вибрати для відображення в робочому вікні одну з існуючих діаграм компонентів або приступити до розробки нової діаграми |
| Deployment Diagram |  | Дозволяє відобразити в робочому вікні діаграму розгортання розробляється моделі |
| Interaction Diagram |  | Викликає діалогове вікно з пропозицією вибрати для відображення в робочому вікні одну з існуючих діаграм кооперації або послідовності , а також приступити до розробки нової діаграми взаємодії |
| State Machine Diagram |  | Викликає діалогове вікно з пропозицією вибрати для відображення в робочому вікні одну з існуючих діаграм станів моделі або приступити до розробки нової діаграми |
| Expand |  | Відображає в робочому вікні першу з діаграм виділеного пакету моделі |
| Parent |  | Відображає в робочому вікні батька виділеної діаграми моделі |
| Specification |  | Викликає діалогове вікно властивостей виділеного елемента моделі |
| Top Level |  | Відображає в робочому вікні діаграму самого верхнього рівня для поточної діаграми моделі |
| Referenced Item |  | Відображає в робочому вікні діаграму класів , яка містить клас для виділеного об'єкта моделі |
| Previous Diagram |  | Відображає в робочому вікні попередню діаграму моделі |
| Create Message Trace Diagram |  | Дозволяє створити діаграму трасування повідомлень |

Спеціальна панель інструментів містить кнопки із зображенням графічних примітивів, необхідних для розробки різних діаграм ( табл. 1.4). Призначення окремих кнопок панелі можна дізнатися також з спливаючих підказок.

Таблиця 1.4 . Призначення кнопок спеціальної панелі інструментів( на прикладі роботи з діаграмою класів)

|  |  |  |
| --- | --- | --- |
| графічне зображення | підказка | призначення кнопки |
|  | Selection Tool | Перетворює зображення курсору у форму стрілки для подальшого виділення елементів на діаграмі |
|  | Text Box | Додає на діаграму текстову область |
|  | Note | Додає на діаграму примітка |
|  | Anchor Note to Item | Додає на діаграму зв'язок примітки з відповідним графічним елементом діаграми |
|  | Class | Додає на діаграму клас |
|  | Interface | Додає на діаграму інтерфейс |
|  | Unidirectional Association | Додає на діаграму спрямовану асоціацію |
|  | Association Class | Додає на діаграму асоціацію клас |
|  | Package | Додає на діаграму пакет |
|  | Dependency or Instantiates | Додає на діаграму відношення залежності |
|  | Generalization | Додає на діаграму відношення узагальнення |
|  | Realize | Додає на діаграму відношення реалізації |

**UML діаграма прецедентів або варіантів використання**

Будь-які системи проектуються з урахуванням того, що в процесі своєї роботи вони будуть використовуватися людьми та/або взаємодіяти з іншими системами. Сутності, з якими взаємодіє система в процесі своєї роботи, називаються Акторами, причому кожен Актор очікує, що система буде вести себе певним передбачуваним чином.

Актор (actor) - це безліч логічно пов'язаних ролей, які виконуються при взаємодії з прецедентами або сутностями (система, підсистема або клас). Актором може бути людина або інша система, підсистема або клас, які представляють щось поза сутністю.

Графічно Актор позначається або "чоловічком" або символом класу з відповідним стереотипом, як показано на рисунку 4. Обидві форми подання мають один і той же зміст і можуть використовуватися в діаграмах.



Рис. 4. Позначення Актора

Прецедент – це опис множини послідовних подій, що виконуються системою, які дають результат Актору. Прецедент представляє поведінку суті, описуючи взаємодію між Актора і системою. Прецедент не показує, "як" досягається певний результат, а тільки "що" саме виконується.

Прецеденти позначаються дуже просто - у вигляді еліпса, всередині якого зазначена його назва (рис. 5.). Прецеденти і Актор з'єднуються за допомогою ліній, які називаються відношенням асоціації (рис. 6.). Відношення асоціації використовується для позначення взаємодії Актора і прецеденту. Часто на одному з кінців лінії зображають стрілку, яка відповідає за розширення прецедентів іншими прецедентами, таке відношення називається узагальненням (рис. 7).



Рис. 5. Позначення прецеденту.



Рис. 6. Позначення відношення асоціації.



Рис. 7. Позначення відношення узагальнення.

Відношення включення (рис. 8.) означає, що в деякій точці базового прецеденту міститься поведінка іншого прецеденту. Включений прецедент не існує сам по собі, а є всього лише частиною базового прецеденту. Таким чином, базовий прецедент ніби запозичує поведінку включених прецедентів.

Відношення розширення (рис. 9.) доповнює прецедент іншими прецедентами, що спрацьовують при деяких умовах, додає у вихідний прецедент послідовність дій, що містяться в іншому прецеденті.



Рис. 8. Позначення відношення включення



Рис. 9. Позначення відношення розширення.

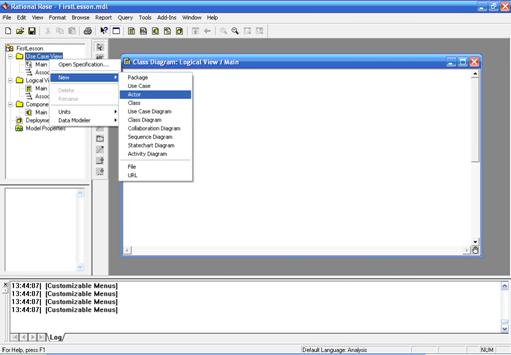
Розглянемо приклад варіантів використання для системи продажу товарів в Інтернет-магазині, що зображений на рисунку 10. Клієнт може переглянути список товарів, змінити вміст корзини, оформити замовлення, оплатити товар та здійснити дзвінок. Прецедент *реєстрація покупця* є тільки частиною прецеденту *оформлення замовлення*. Прецедент *оплата товару* може включати прецедент *оплата кредиткою* або *готівкою*. Прецедент *міський дзвінок* є уточненням прецеденту *виконати* *телефонний дзвінок*.



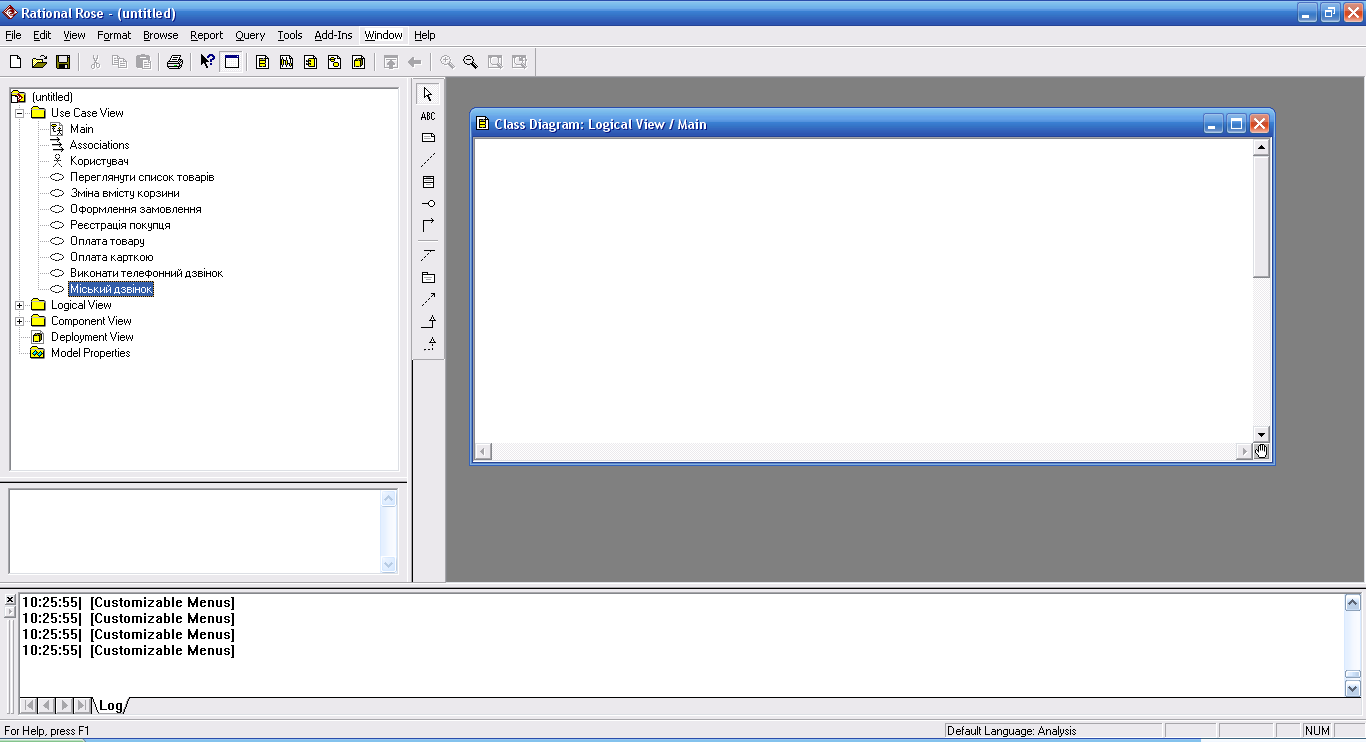
Рис. 10. UML-діаграма варіантів використання системи продажу товарів в Інтернет-магазині

**Створення UML діаграми прецедентів або варіантів використання в програмі Rational Rose**

Для цього потрібно створити порожній проект, переключитися на папку Use Case View - для роботи з діаграмами прецедентів (Use case) і відкрити контекстне меню натисненням правої кнопки миші. Якщо вибрати пункт New/Actor (рис. 11), то ви отримаєте дійова особа.



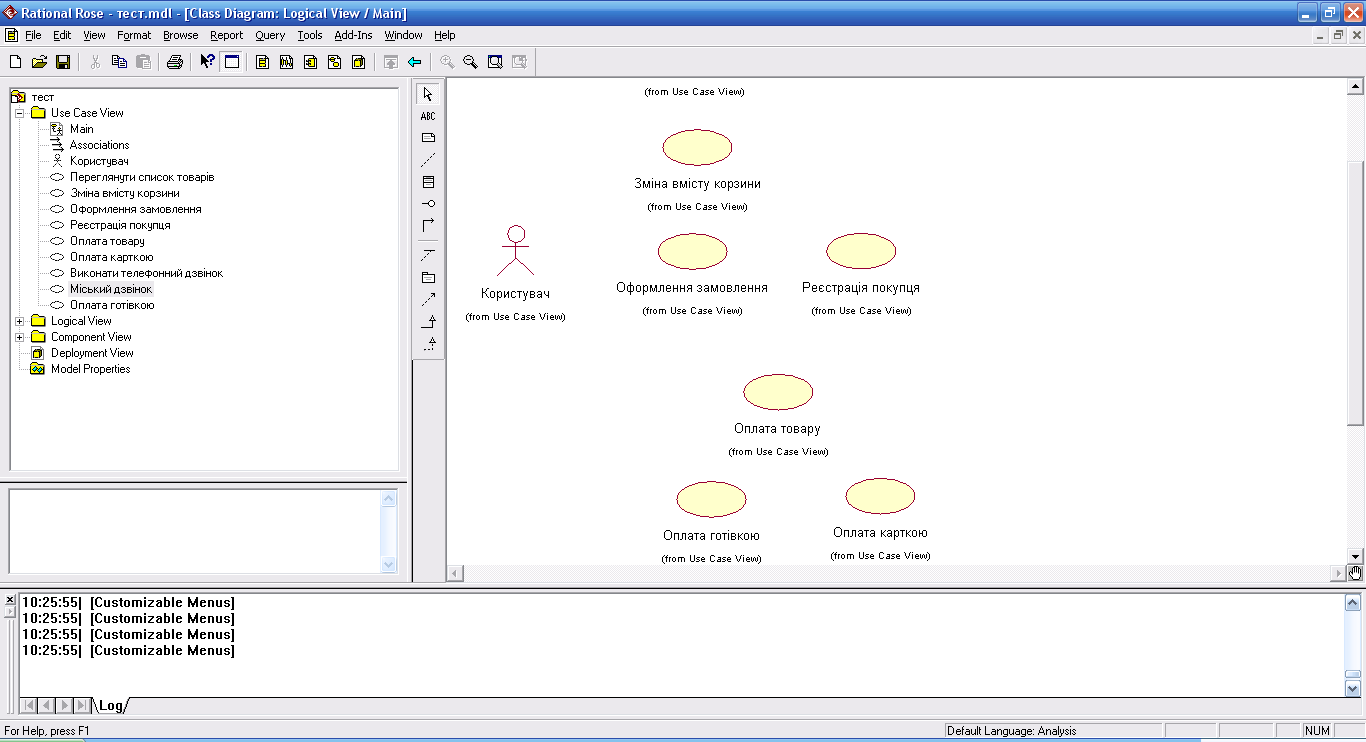
Тепер створимо всі описані вище прецеденти (рис.12)



Всі введені дійові особи і сценарії поведінки негайно з'являються у вікні браузера, звідки можна перетягувати їх мишею на діаграми.

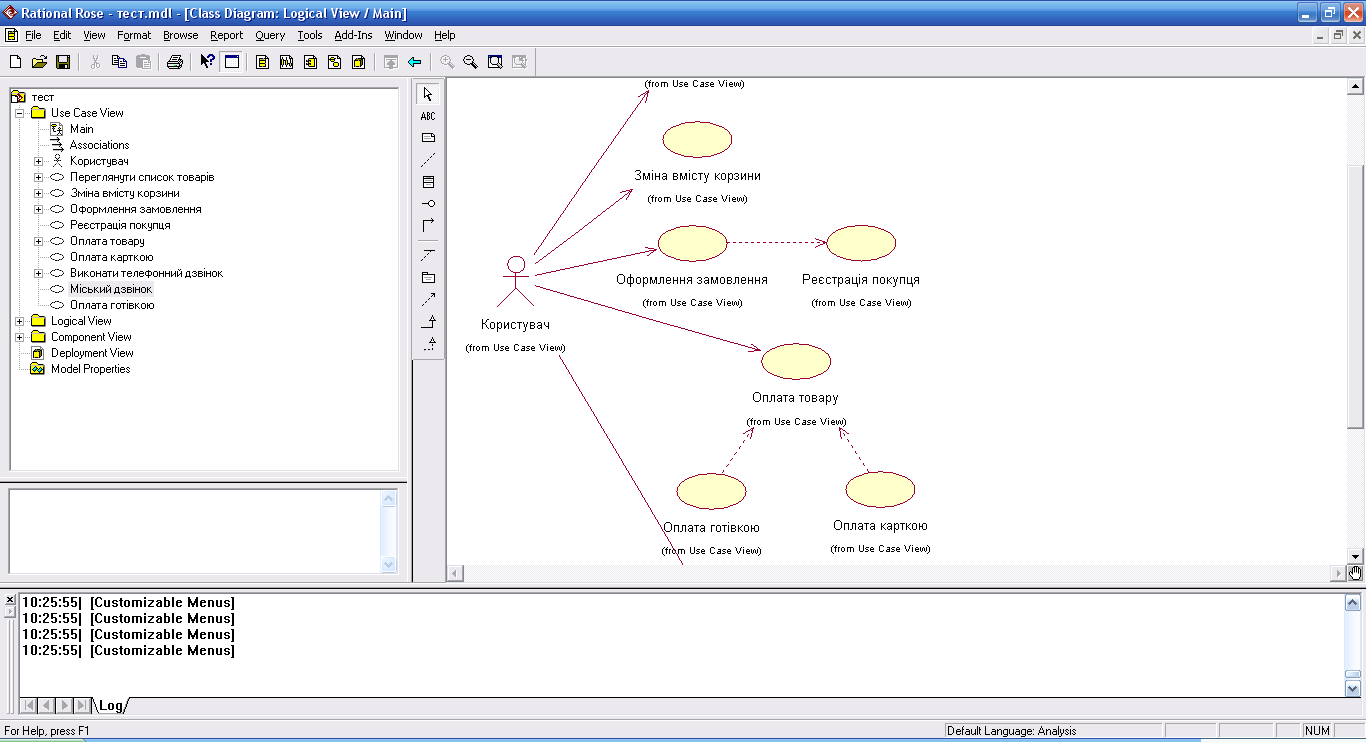
Далі, як правило, будується діаграма сценаріїв поведінки - прецедентів. Для цього подвійним клацанням на піктограмі Main з папки Use Case View відкривається головна діаграма сценаріїв.

У неї з вікна браузера перетягуються всі дійові особи і сценарії поведінки, які були створені в рамках моделі. (Рис. 13)

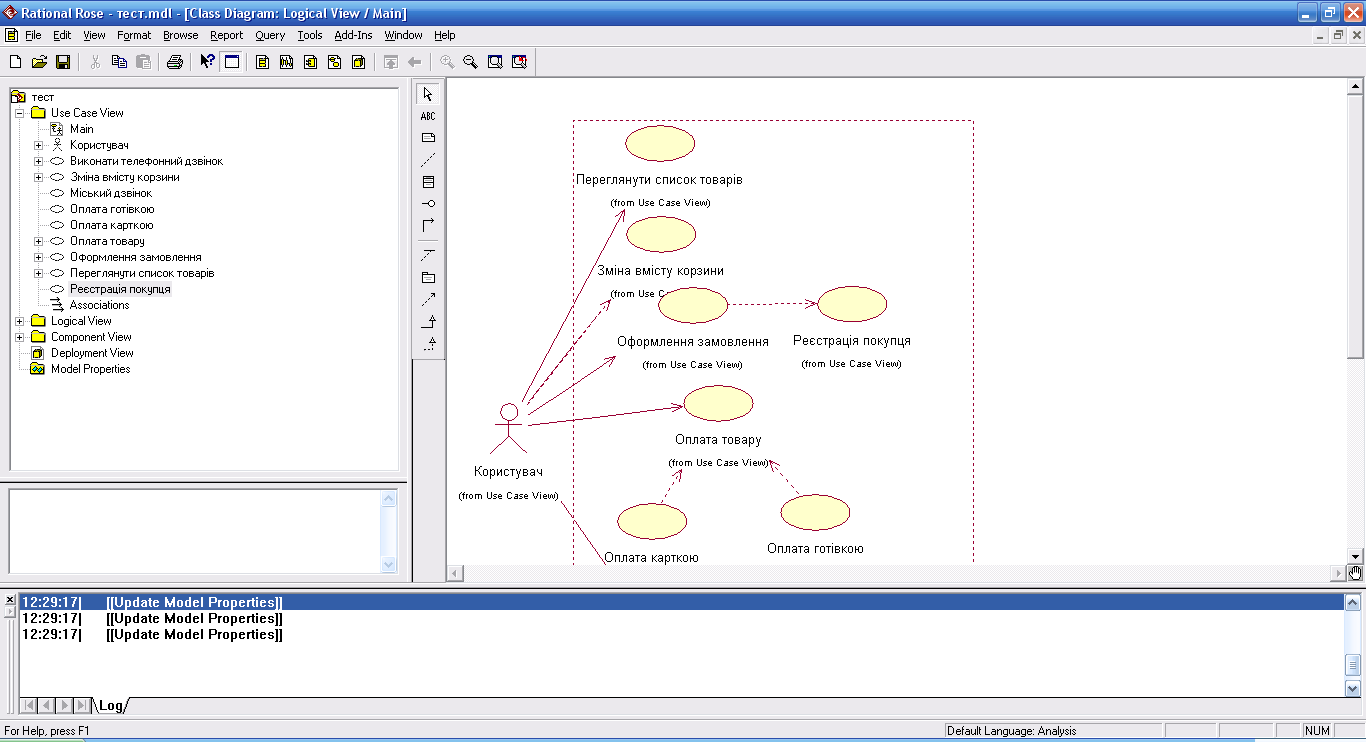


Після розміщення, ці компоненти потрібно зв'язати між собою, щоб відобразити взаємозв'язки. У моделі найкраще підійдуть зв'язки типу «однонаправлена ​​асоціація» (Unidirectional Association) .

Для реалізації зв'язків застосовується метод перетягування. Спочатку в палітрі вибирається тип зв'язку «однонаправлена ​​асоціація» (Uniderectional Association ), після чого потрібно протягнути лінію між дійовою особою і сценарієм поведінки . У результаті на діаграмі виникне стрілка . Аналогічним чином поступають з усіма компонентами діаграми . Готова діаграма показана на Рис. 14 .



Для створення кордону («інтерфейсу») між акторами і прецендентом можна скориставшись пунктом меню «Tools/Create/Note Anchor » або вибрати серед піктограм інструмент «Anchor Note to Item». Вибравши інструмент необхідно клацнути один раз на середині однієї зі стрілок відносини далі у всіх кутах інтерфейсу і завершити кордон у місці початку її малювання . Якщо прямокутник вийшов не дуже рівний то це можна виправити вибравши в меню « Format / Line Style / Rectilinear»



Завдання

Розробити в програмі Rational Rose UML діаграми варіантів використання автоматизованої інформаційної системи згідно варіанту.

Варіанти

1. Інформаційно-пошукова система для обліку пацієнтів в поліклініці.
2. Інформаційно-пошукова система для обліку диспансерних хворих в поліклініці.
3. Інформаційно-пошукова система для обліку пацієнтів з інвалідністю в поліклініці.
4. Інформаційно-пошукова система для обліку кадрів у навчальному закладі.
5. Інформаційно-пошукова система для обліку та наявності ліків у аптеці.
6. Сайт інформаційно-довідкової системи пошуку житла.
7. Інформаційно-пошукова система для обліку клієнтів у ветеринарній клініці.
8. Сайт дистанційного навчання студентів.
9. Web-сайт по наданню сервісу зберігання файлів.
10. Інформаційна системи для діяльності банку.
11. Інформаційна системи для диспетчера автобусного парку.
12. Інформаційна система для пункту прокату спортивного спорядження.
13. Сайт туристичної фірми.
14. Інформаційна система готельного бізнесу.
15. Розробка тестувальної програми для студентів.