МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

ІНСТИТУТ ПІСЛЯДИПЛОМНОЇ ОСВІТИ

КАФЕДРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

****

**Звіт**

**з лабораторної роботи №1**

# на тему: «Методологія моделювання програмного забезпечення засобами UML, на прикладі інформаційної системи обліку товарів на скаладі»

***з дисципліни*** ***“Моделювання програмного забезпечення”***

**ВИКОНАВ:**

Гринчук Т.А.

ПЗС - 21

**ПЕРЕВІРИВ:**

доц. Яковина В.С.

Львів - 2014

**Мета:** Ознайомлення з основними елементами визначення, подання, проектування і моделювання програмних систем за допомогою мови UML.

**Теоретичні відомості**

Графічним представленням предметної області засобами UML (Unified Modeling Language- уніфікована мова моделювання) є діаграма класів.

Діаграми класів – центральна ланка об’єктно-орієнтованих методів розробки програмного забезпечення.

UML пропонує використати три рівні діаграм класів залежно від ступеня їхньої деталізації:

• концептуальний рівень, на якому діаграми класів, названі в цьому випадку контекстними, демонструють зв'язки між основними поняттями предметної області;

• рівень специфікацій, на якому діаграми класів відображають інтерфейси класів предметної області, тобто зв'язки об'єктів цих класів;

• рівень реалізації, на якому діаграми класів безпосередньо показують поля й операції конкретних класів.

Основним поняттям у моделі ставляться у відповідність класи. Клас при цьому традиційно розуміють як сукупність загальних ознак заданої групи об'єктів предметної області. Відповідно до цього визначення на діаграмі класів кожному класу відповідає група об'єктів, загальні ознаки яких фіксує клас.

Позначення класу в UML – прямокутник, розділений на 3 частини: назва класу, атрибути, операції.

Атрибут - це поіменована властивість класу, що включає опис множини значень, які можуть приймати екземпляри цієї властивості. В якості атрибутів представляють деякі, істотні з погляду розв'язуваного завдання характеристики об'єктів, наприклад ідентифікуючі значення (ім'я, номер). Для конкретного об'єкта атрибут завжди має певне значення.

Операція - це абстракція того, що дозволено робити з об’єктом. У всіх об’єктів класу є спільний набір операцій. Клас може містити довільне число операцій або не містити їх взагалі. Часто звернення до операції об’єкта змінює його стан або його дані. Операції класу зображаються в розділі, розміщеному нижче розділу з атрибутами. При цьому можна обмежитися тільки іменами.

При побудові абстракцій класи рідко існують автономно. Як правило, вони різними способами взаємодіють між собою.

Існує три види відношень, особливо важливих для об’єктно-орієнтованого моделювання:

залежності, які описують існуючі між елементами системи відношення використання;

узагальнення, які зв’язують узагальнені класи з спеціалізованими;

асоціації, які представляють структурні відношення між об’єктами.

Кожне з цих відношень дозволяє по-різному комбінувати сутності.

Відношення залежності використовується в такій ситуації, коли деяка зміна одного елемента моделі вимагає зміни іншого, залежного від нього елемента моделі. Це відношення зображається штрих-пунктирною лінією із звичайною стрілкою в сторону незалежного елемента (від елемента-клієнта до елемента-джерела).

Відношення узагальнення на діаграмі класів описує ієрархічну будову класів і успадкування їх властивостей та поведінки. При цьому похідний (спеціалізований) клас має всі властивості та поведінку базового (узагальненого) класу, а також свої власні властивості та поведінку, які відсутні у базовому класі. Це відношення зображається суцільною лінією з трикутною стрілкою в сторону базового елемента (від потомка до батька).

Відношення асоціації відповідає наявності деякого зв’язку між елементами моделі. Асоціація поєднує рівноправні сутності. Таке відношення зображається суцільною лінією і при потребі доповнюється спеціальними символами. Спеціальні символи характеризують окремі властивості конкретної абстракції.

В якості цих символів можуть бути 1) ім’я асоціації, 2)заповнена всередині стрілка-трикутник, яка вказує на порядок зв’язку, 3) імена елементів-ролей асоціації, 4) кратність цих ролей.

Частковим випадком відношення асоціації є відношення агрегації, яке в свою чергу має спеціальну форму – відношення композиції.

Відношення агрегації має місце між деякими елементами в тому випадку, якщо один елемент є сутністю, яка включає в собі інші сутності в якості складових частин. Дане відношення має фундаментальне значення для опису структури складних систем, оскільки застосовується для представлення системних взаємозв’язків типу «частина-ціле». Таким чином утворюється деяка ієрархія, яка принципово відрізняється від тої, що породжується відношенням узагальнення. Частини системи ніяк незобов’язані наслідувати властивості і поведінку системи, оскільки є самостійними сутностями. Більш того, частини цілого мають свої атрибути і операції, які істотно відрізняються від атрибутів і операцій всього цілого. Графічно відношення агрегації зображається суцільною лінією, один з кінців якої є незаповненим всередині ромбом. Цей ромб вказує на той елемент, який «цілим» (системою).

Відношення композиції (строга агрегація) є частковим випадком відношення агрегації. Специфіка цього взаємозв’язку заклечається в тому, що частини не можуть виступати у відриві від цілого. Тобто із знищенням цілого знищуються і всі його складові частини. Відношення композиції зображається суцільною лінією, один з кінців якої є заповненим всередині ромбом. Цей ромб вказує на той елемент, який «цілим» (системою).

**Виконання роботи**

У прикладному рішенні «1С-Логістика: Управління складом» реалізований детальний оперативний складський облік руху товарів. Забезпечується складський облік запасів (повний контроль запасів товарів на підприємстві).

Організація обліку товарів дозволяє:

* управляти залишками товарів у різних одиницях виміру на безлічі складів;
* вести роздільний облік власних товарів, товарів, прийнятих і переданих на реалізацію;
* деталізувати розташування товару на складі по місцях зберігання, що дозволяє оптимізувати збірку товарів на складі;
* враховувати серії товарів (серійні номери, терміни придатності і т. д.);
* задавати довільні характеристики товарів (колір, розмір і т. д.);
* враховувати ВМД і країну походження;
* оформляти операції зборки / розбирання товарів;

резервувати товари

Очевидно, що в системі обліку товарів на складі повинні зберігатися не тільки нормативи, тарифи й інформація про послуги, але й дані про клієнтів, укладені контракти з абонентами з сторонніми постачальниками послуг зв'язку (якщо мережа даного оператора пов'язана з іншими), а також про вартість передачі інформації з різних каналів і напрямків. Крім того, будь-яка розрахункова система неможлива без «історії» платежів і виставлених рахунків усім клієнтам, оскільки саме ці відомості дозволяють організувати контроль над оплатою й автоматизувати так звану активацію/деактивацію абонентів. Чим могутніше виконавчий механізм СКБД, тим більше масштабною та багатофункціональною буде білінгова система, побудована на її основі.

Фукціонал даного продукту реалізовує вирішення наступних проблем складського господарства:

* Затримки при виконанні складських операцій
* «Втрати» товару на складі
* Втрати пов'язані з пересортицею
* Втрати пов'язані із закінченням терміну придатності
* Висока складність проведення інвентаризації
* Залежність від персоналу складу

«1С-Логістика: Управління складом» реалізовує такі функції:

***оптимізація:***

* Використання складських площ
* використання обладнання
* Використання трудових ресурсів

***скорочення***

* Часу на проведення всіх складських операцій
* Помилок при виконанні складських операцій
* Витрат на заробітну плату складських працівників
* Витрат на складське зберігання
* Втрат товару (терміни придатності, злодійство і т.д.)
* Підвищення точності обліку ТМЦ

Типова схема документообороту має вигляд (рис. 1):



Рис. 1. Документооборот системи

Основні функції системи

* визначення топології складського комплексу
* приймання
* контроль якості
* розміщення
* підбір для відвантаження
* відвантаження
* внутрішньоскладські переміщення
* інвентаризація
* списання
* перепакування
* перехресна відвантаження
* Формування аналітичної звітності
* підтримка роботи з торговим обладнанням
* контроль роботи персоналу
* інтеграція з КІС
* функції комерційного складу
* управління правами доступу

Топологія, занесена в БД, є всього лише віртуальною копією складу матеріального. Основними об'єктами, регулюючими топологію складу в інформаційній системі, є довідники «Склади» і «Осередки» (рис. 2):

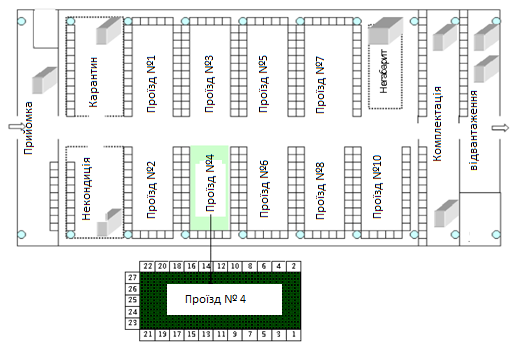


Рис. 2. Топологія складу

У режимі користувача для місць зберігання задається (рис. 3):

* адресація
* кількість
* типорозміри
* розташування
* правила роботи з товаром

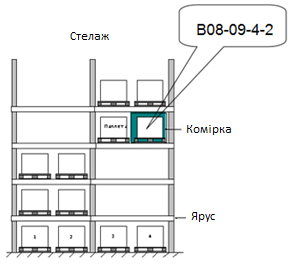
****

Рис. 3. Розмітка комірок

Вхідний потік товарів в системі та розміщення на складі складається з таких етапів (рис. 4):

* Початок планування розміщення:
* Після приймання всієї поставки
* Після приймання частині поставки - на прийняту частина
* Спосіб планування розміщення:
* Автоматично . системний запуск
* Автоматично . користувальницький запуск
* Вручну попередньо
* Не планувати - вносити в систему за фактом

Завдання правил розміщення:

* Область розміщення для кожного типу упаковки
* Розміщення товару у вільні комірки
* Розміщення товару в частково зайняті осередки
* Регульований контроль змішування товарів в комірці
* Контроль за вагою й обсягом

Поповнення зони відбору

* Завдання маршруту пошуку місця для системи
* Завдання маршруту обходу місць - для комірників

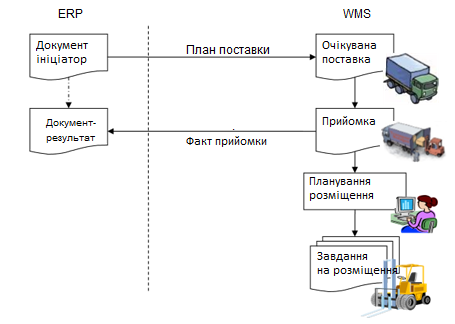
****

Рис. 4. Вхідний потік товарів

Завантаження даних з облікової системи:

* Планування приймання
* Рекомендується, але не обов'язково
* Ідентифікація та маркування товару
* Приведення товару до стандарту складського зберігання
* Приймання фактичного надходження за кількістю та якістю

Введення додаткової інформації:

* терміни придатності
* серійні номери
* ....
* Резервування прийнятого товару під замовлення (крос-докинг)
* План-фактний аналіз приймання, коректування планового документа
* Друк стандартних форм складських документів надходження
* Вивантаження даних в облікову систему

Вихідний потік товарів має вигляд (рис. 5):

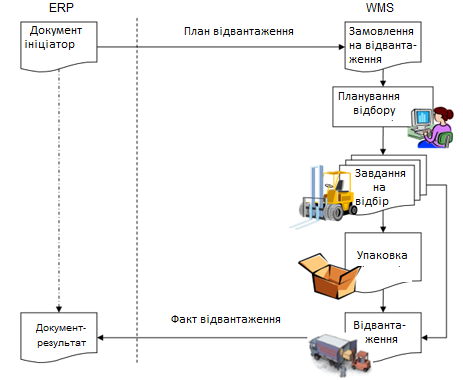
****

Рис. 5. Вихідний потік товарів

Основні властивості інвентаризації в системі:

* Множинний фільтр відбору осередків для інвентаризації:
  + по товару
  + За кількістю залишку
  + з проблеми
  + за заповнювання
  + по розташуванню
* Графік інвентаризації осередків
* Блокування інвентаризуються осередків

В системі можна отримати наступні звіти по залишках товарів:

* У розрізі місць зберігання
* У розрізі товарної аналітики
* У розрізі замовлень на відвантаження
* Рух товарів
* Аналіз виконання замовлень
* Аналіз роботи співробітників
* Стан складу
* Заповненість осередків
* Контроль приймання
* Контроль відвантаження
* Плани витрат

UML-діаграма основних класів програмного продукту зображена на рис. 6:



Рис. 6. UML-діаграма класів

Як видно з діаграми, на етапі проектування розробники виділяють батьківські класи «Довідники» та «Документи», які будуть мати атрибути та методи, що наслідуються для дочірніх довідників: Товари, Контрагенти та Склади, а також документів: ПрихіднаНакладна, ВидатковаНакладна. Також, окремо виписуємо глобальний об’єкт ЗалишкиТоварів, що залежить від зміни документів ПрихіднаНакладна та ВидатковаНакладна. На цьому етапі ми також проектуємо можливість довільного виду документа мати скільки завгодно стрічок.

Процес опрацювання документації користувачем зручно зобразити на наступній діаграмі прецендентів (рис. 7):



Рис. 7. Діаграма прецендентів

Як бачимо, типовий користувач системи (не з повними правами) має можливості: проводити документи, друкувати їх та видаляти. Кожне проведення веде за собою перевірку залишку та чи дата документа оперативна. Якщо одна з цих перевірок не виконалась – документ не проводиться. Видаляти також можна лише документи оперативно (тільки поточної дати), інакше не виконається перевірка на дату документа і видалення не відбудеться. При проведенні є можливість також роздрукувати макет документа (якщо цього захоче користувач).

Діяльність відповідального менеджера складу по роботі з замовленнями, які поступають на склад, можна зобразити на наступній схемі діяльності (рис. 8).

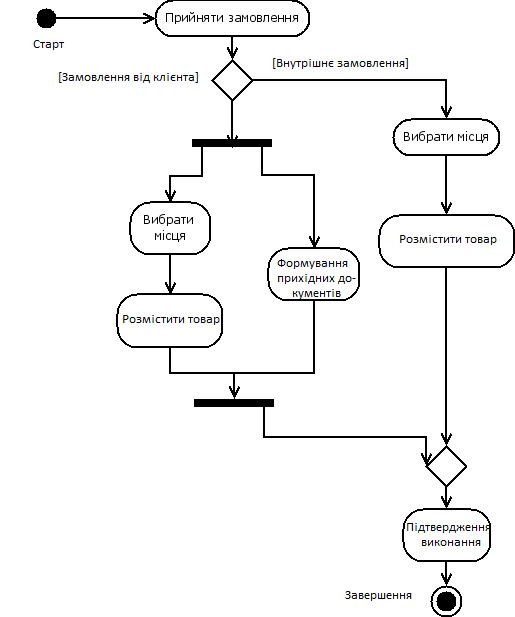


Рис. 8. Діаграма діяльності менеджера складу по роботі з замовленнями

Як бачимо процес відрізняється, в залежності, чи на склад прийшло внутрішнє замовлення на розміщення власного товару організації, чи від стороннього контрагента. При сторонньому замовленні менеджер розміщає товар на складі та паралельно формується пакет необхідних документів у відділі документообороту складу.

**Висновок**

У даній лабораторній роботі я виконав моделювання програмного забезпечення на прикладі інформаційної системи обліку товарів на складі. Було розроблено такі види діаграм на мові UML: діаграма класів (Class diagram), у якій зображено набір класів; діаграма прецедентів (Use Case Diagram), яка зображає сценарії взаємодії між акторами (користувачами) і прецедентами; діаграма діяльності (Activity diagram); діаграма послідовностей (Sequence diagram), яка описує поведінкові аспекти системи. Також я закріпив на практиці знання уніфікованої мови моделювання UML.