

Специфікація функціональних вимог

Проектування інформаційних систем

Лекція 1

Глазунова О.Г.,
д.п.н., проф.

1. Аналіз вимог
2. Виявлення та документування
3. Варіанти специфікації
4. Функціональні вимоги
5. Кейси

План

Посилання на ЕНК:

<https://elearn.nubip.edu.ua/course/view.php?id=231>



1. Аналіз вимог

Етап розробки ІС

Життєвий цикл ПЗ (ЖЦ) - неперервний процес з моменту прийняття рішення про створення ПЗ до вилучення його з експлуатації.

Найменування документа (укр.)	ДСТУ ISO/IEC/IEEE 12207:2018 Інженерія систем і програмних засобів. Процеси життєвого циклу програмних засобів (ISO/IEC/IEEE 12207:2017, IDT)
Дата початку дії	15.08.2018
Дата прийняття	06.08.2018
Статус	Діючий
Мова документа	Англійська
На заміну	ДСТУ ISO/IEC 12207:2016
Затверджуючий документ	Наказ від 06.08.2018 № 261 Про прийняття та скасування національних нормативних документів
Вид документа	ДСТУ (Державний Стандарт України)
Шифр документа	12207:2018
Розробник	ДП «Український науково-дослідний і навчальний центр проблем стандартизації, сертифікації та якості» (ДП «УкрНДНЦ»)
Орган, що прийняв	ДП «Український науково-дослідний і навчальний центр проблем стандартизації, сертифікації та якості» (ДП «УкрНДНЦ»)

Документ відповідає офіційному тексту.

З питань придбання офіційного видання звертайтеся до національного органу стандартизації (ДП «УкрНДНЦ»)

ДСТУ ISO/IEC/IEEE 12207:2018

- Інженерія систем і програмних засобів. Процеси життєвого циклу програмних засобів

Розділи стандарту ДСТУ ISO/IEC/IEEE 12207:2018)

- 1 Scope
- 2 Normative references
- 3 Terms, definitions, and abbreviated terms
- 4 Conformance
- 5 Key concepts and application
 - 5.1 Introduction
 - 5.2 Software system concepts
 - 5.3 Organization and project concepts
 - 5.4 Life cycle concepts
 - 5.5 Process concepts
 - 5.6 Process groups
 - 5.7 Process application
 - 5.8 Process reference model
- 6 Software life cycle processes
 - 6.1 Agreement processes
 - 6.2 Organizational Project-Enabling processes
 - 6.3 Technical Management processes
 - 6.4 Technical processes

Успішні ~40

Проблемні ~40

Провальні ~20

ІТ-проекти

The Standish Group CHAOS Report 2011-2019



1

Успіх

- Підключення до розробки користувача (15,9%)
- Підтримка з сторони виконавчого керівництва (13,9%)
- Чітка постановка вимог (13,0%)

2

Провал

- Недостатньо вихідної інформації від клієнта (12,8%)
- Неповні вимоги і специфікації (12,3%)
- Зміни вимог і специфікацій (11,8%)

PROJECT SUCCESS RATES

AGILE VS WATERFALL



WWW.VITALITYCHICAGO.COM

Source: Standish Group Chaos Studies 2013-2017

Бізнес- вимоги

Вимоги користувачів

Функціональні вимоги

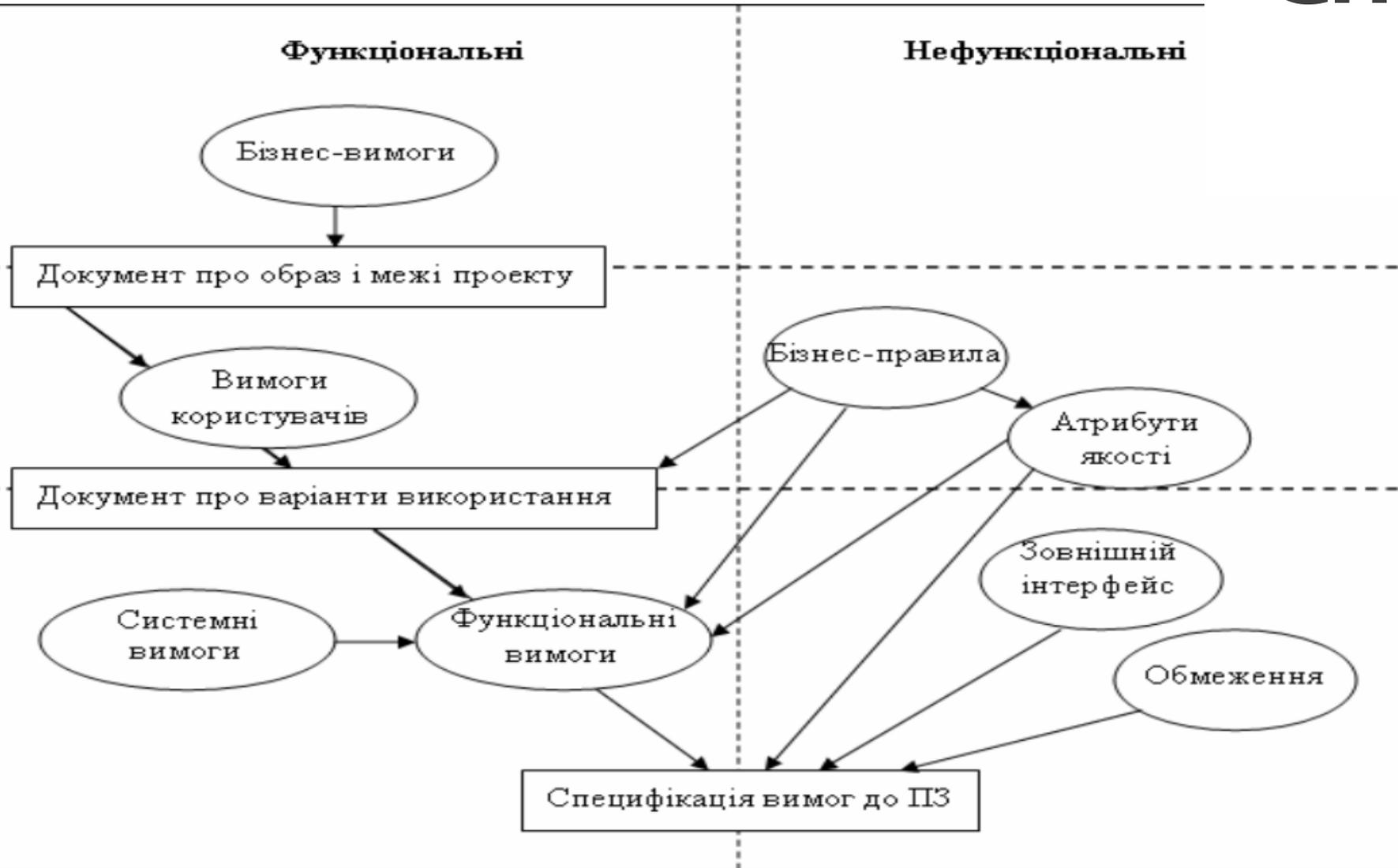
Нефункціональні вимоги

Рівні вимог

Визначення

Це описи поведінки системи, властивості системи або її атрибути. Вони можуть бути обмежені процесом розробки системи. (Sommerville,1997).

Специфікація ВИМОГ



Повнота

Коректність

Здійсненність

Необхідність

Пріоритетність

Недвозначність

Можливість перевірки

Характеристики вимог

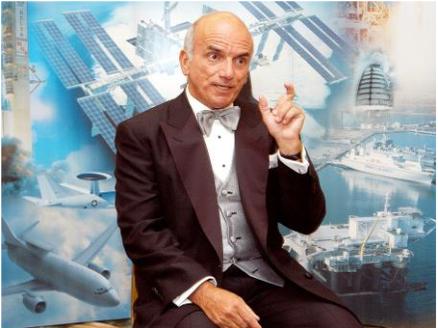
Набір вимог до специфікації:

- Повнота
- Узгодженість
- Модифікація
- Трасованість



2. ВИЯВЛЕННЯ ВИМОГ

Документування вимог



Замовник



Артефакти



«Кращі
практики»

Список документов регистрации программных продуктов контрагента

№	Номер	Партнер	Контрагент	Комментарий	Оплат
08.10.2013 19:02:18	0000000024				Алато
30.12.2013 18:36:26	0000000025	партнер	Вертекс		Алато
01.01.2014 0:00:00	0000000017	Покупатель №1	Покупатель №1		Алато
10.01.2014 14:30:48	0000000026				Алато
10.01.2014 14:31:02	0000000027	партнер	ВДГБ: Москва		Алато
16.01.2015 10:33:55	0000000028	партнер	Лагеласский за...		Алато

и, присоединенные к документу

№	Наименование	Описание	Дата со
---	--------------	----------	---------



Джерела вимог

Інтерв'ю

Анкетування

Спостереження

Самостійний опис

JAD

Прототипування

Додайте нижній колонтитул

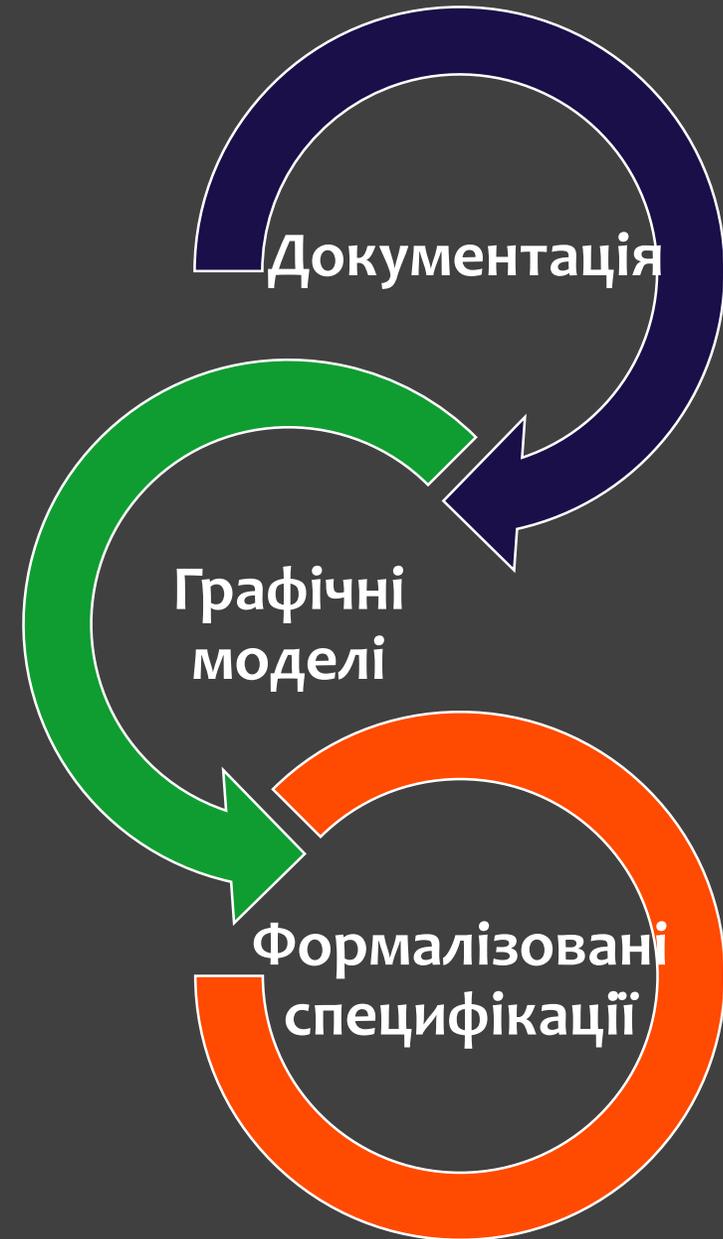
Стратегії

Виявлення вимог

3. Варіанти специфікації



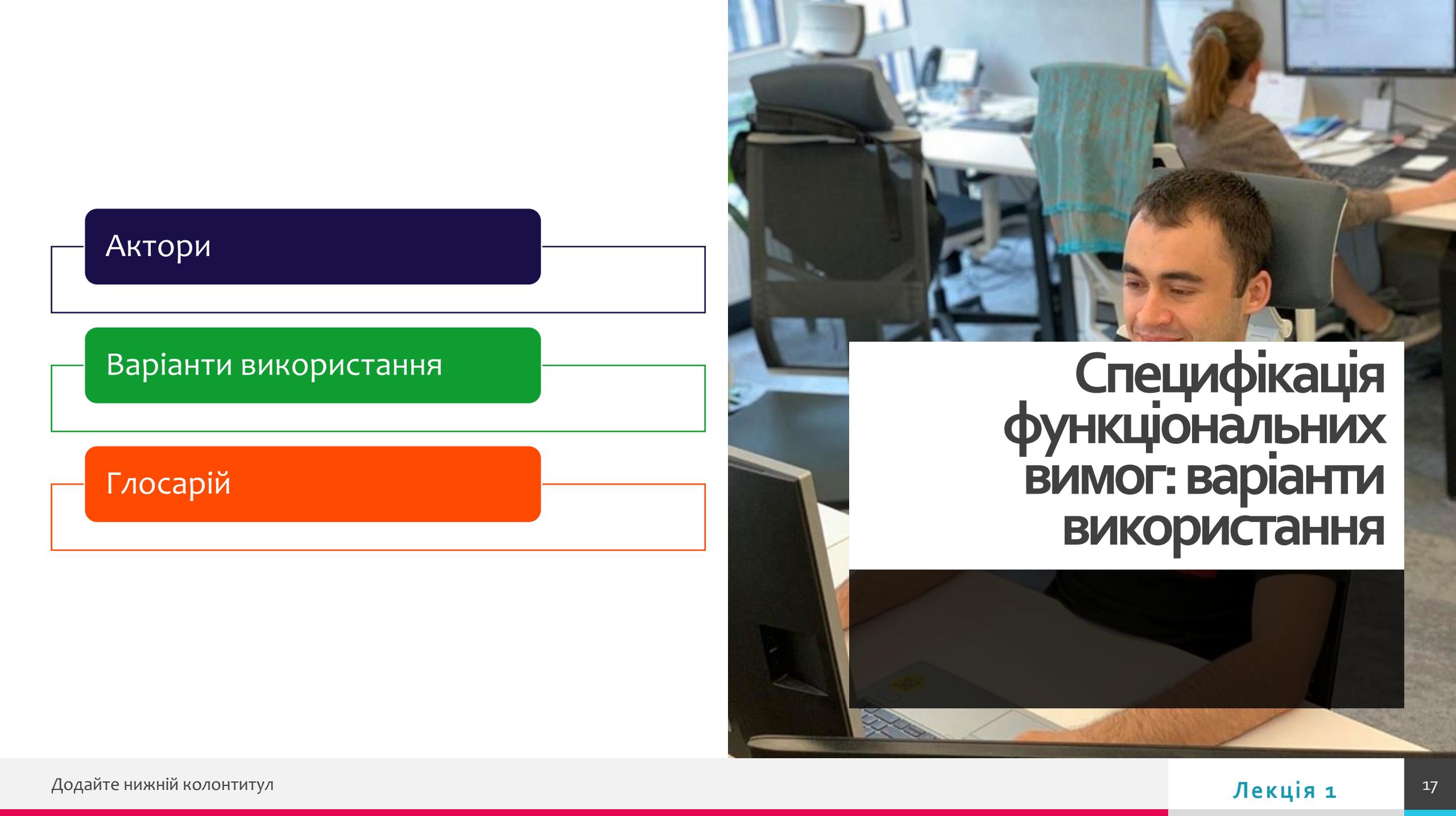
Способи представлення вимог



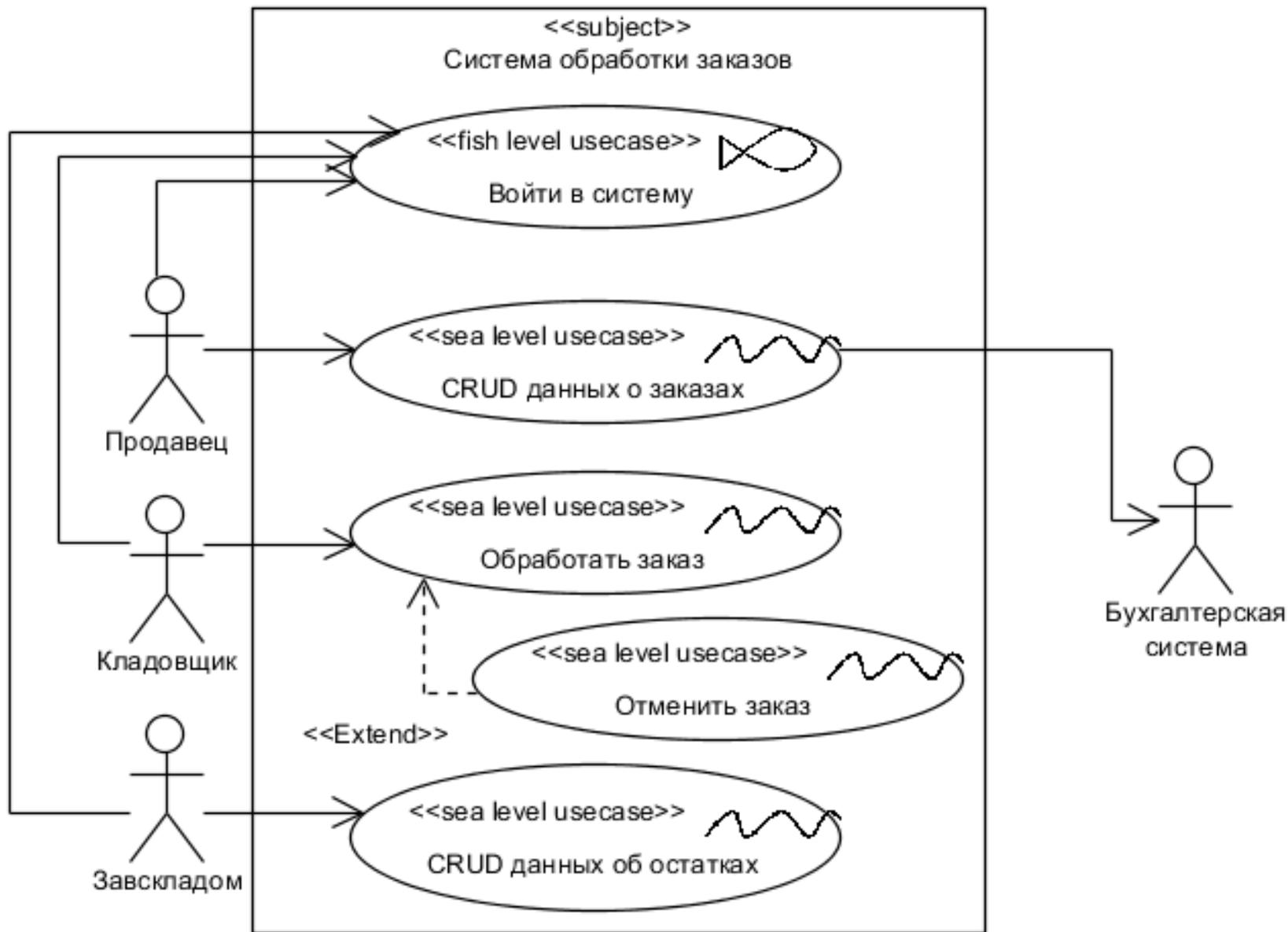
Актори

Варіанти використання

Глосарій



Специфікація функціональних вимог: варіанти використання



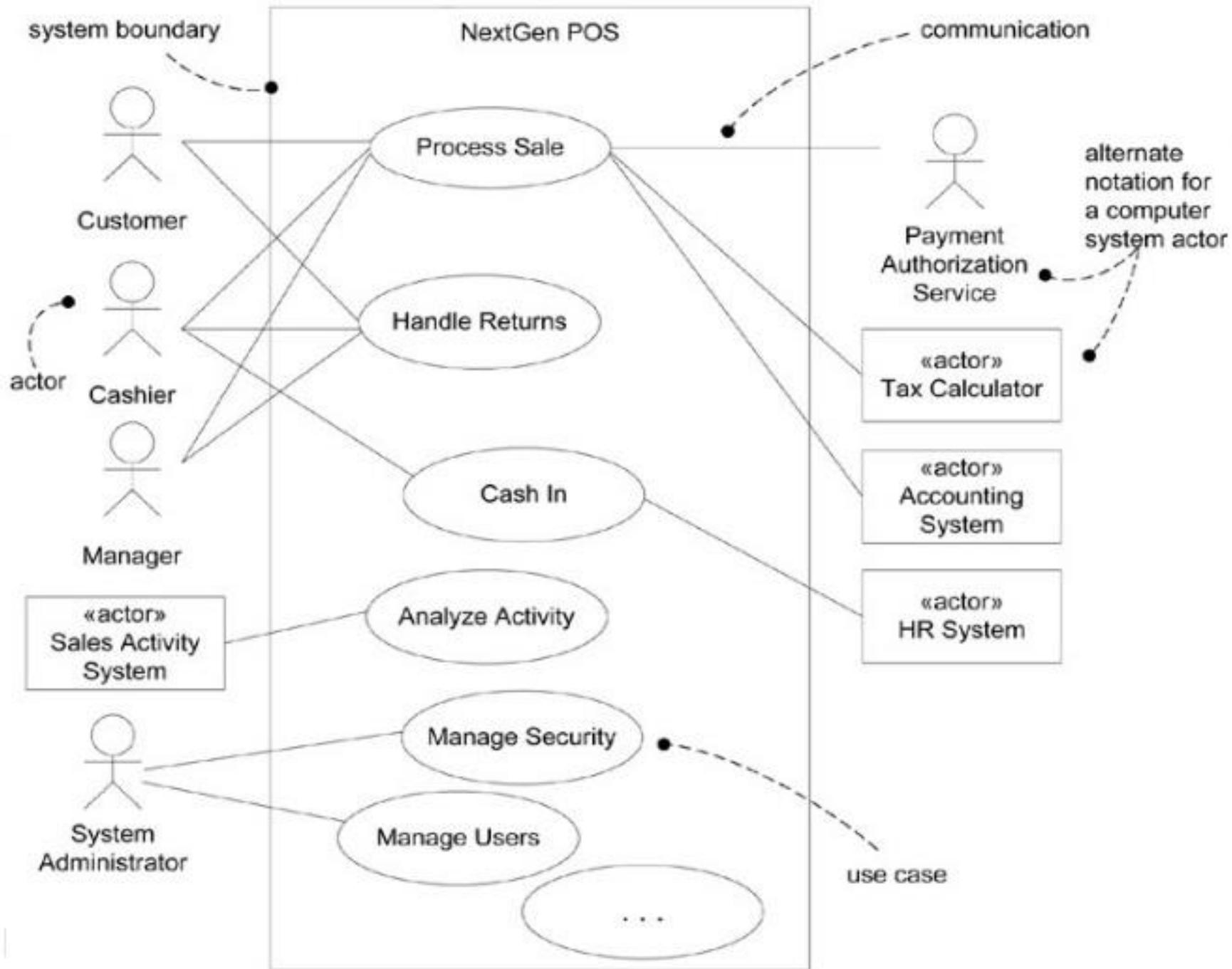
Актор

Поведінка системи

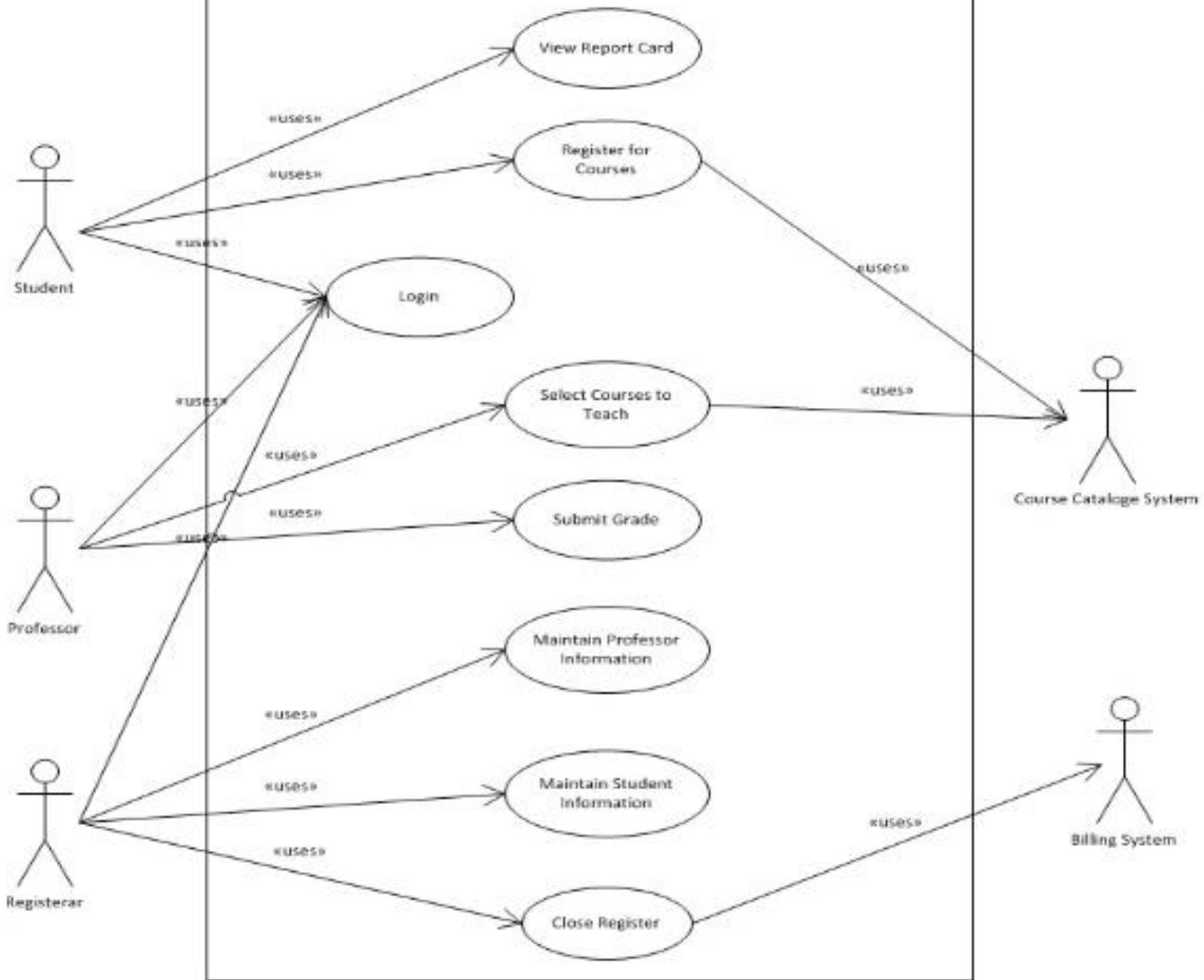
- Компанія по прокату авто
- Система продажу
- Платіжна система
- Система управління навчальними курсами

Варіанти використання

Приклади



Система продаж



Система управління курсами

Пример 1. Разблокировать учетную запись пользователя (простой короткий пример, без альтернативного потока событий):

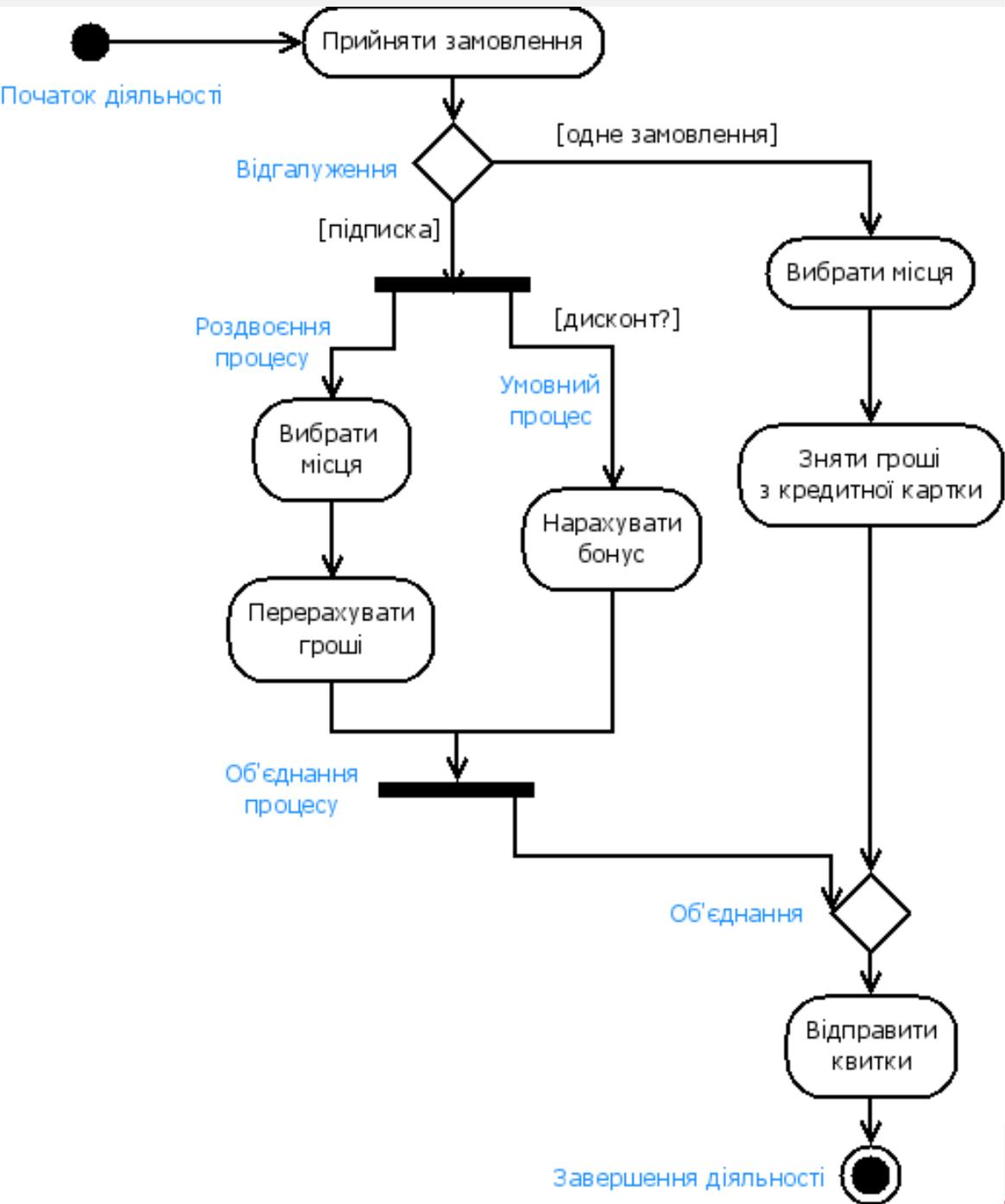
Действующие лица	Администратор, Система
Цель	Изменить статус учетной записи пользователя на «активный».
Предусловие	Учетная запись пользователя не активна.
Успешный сценарий:	<ol style="list-style-type: none">1. Администратор выбирает пользователя и активирует «Разблокировать».2. Система переключает учетную запись пользователя в статус «активный», и посылает сообщение (<i>тут можно сослаться на текст сообщения из списка сообщений, см. примечание ниже</i>) пользователю на email (если «User Account → email» не пусто).
Результат	Учетная запись пользователя была переведена в статус «активный».

Пример 2. Авторизация пользователя:

Действующие лица	Пользователь, Система
Цели	Пользователь: авторизоваться в системе и начать работать; Система: идентифицировать пользователя и его права.
Успешный сценарий:	<ol style="list-style-type: none">1. Пользователь запускает систему. Система открывает сессию пользователя, предлагает ввести логин и пароль.2. Пользователь вводит логин и пароль.3. Система проверяет логин и пароль.4. Система создает запись в истории авторизаций (IP адрес пользователя, логин, дата, рабочая станция).5. Система выдает пользователю сообщение по поводу успешной авторизации (<i>ссылка на сообщение</i>).



Сценарій



Діаграма діяльності

- Під час моделювання поведінки проектованої або аналізованої системи виникає необхідність не тільки подати процес зміни її станів, але й деталізувати особливості алгоритмічної і логічної реалізації виконуваних системою операцій.

ДІАГРАМА ДІЯЛЬНОСТІ (ACTIVITY DIAGRAM)

- У контексті мови UML діяльність (activity) є деякою сукупністю окремих обчислень, що виконуються автоматом.
- Окремі елементарні обчислення можуть приводити до деякого результату або дії (action).
- На діаграмі діяльності відображається логіка або послідовність переходу від однієї діяльності до іншої, при цьому увага фіксується на результаті діяльності.

ДІАГРАМА ДІЯЛЬНОСТІ (ACTIVITY DIAGRAM)

- Є спеціальним випадком стану з деякою вхідною дією і принаймні одним переходом, що виходить зі стану.
- Цей перехід неявно припускає, що вхідна дія вже завершилася.
- Стан дії не може мати внутрішніх переходів, оскільки він є елементарним.
- Звичайне використання стану дії полягає в моделюванні одного кроку виконання алгоритму (процедури) або потоку керування.

Стан дії (action state)

- Графічно стан дії зображається фігурою, що нагадує прямокутник, бічні сторони якого замінені опуклими дугами. Всередині цієї фігури записується вираз дії (action-expression), яка має бути унікальною в межах однієї діаграми діяльності.



Стан дії (action state)

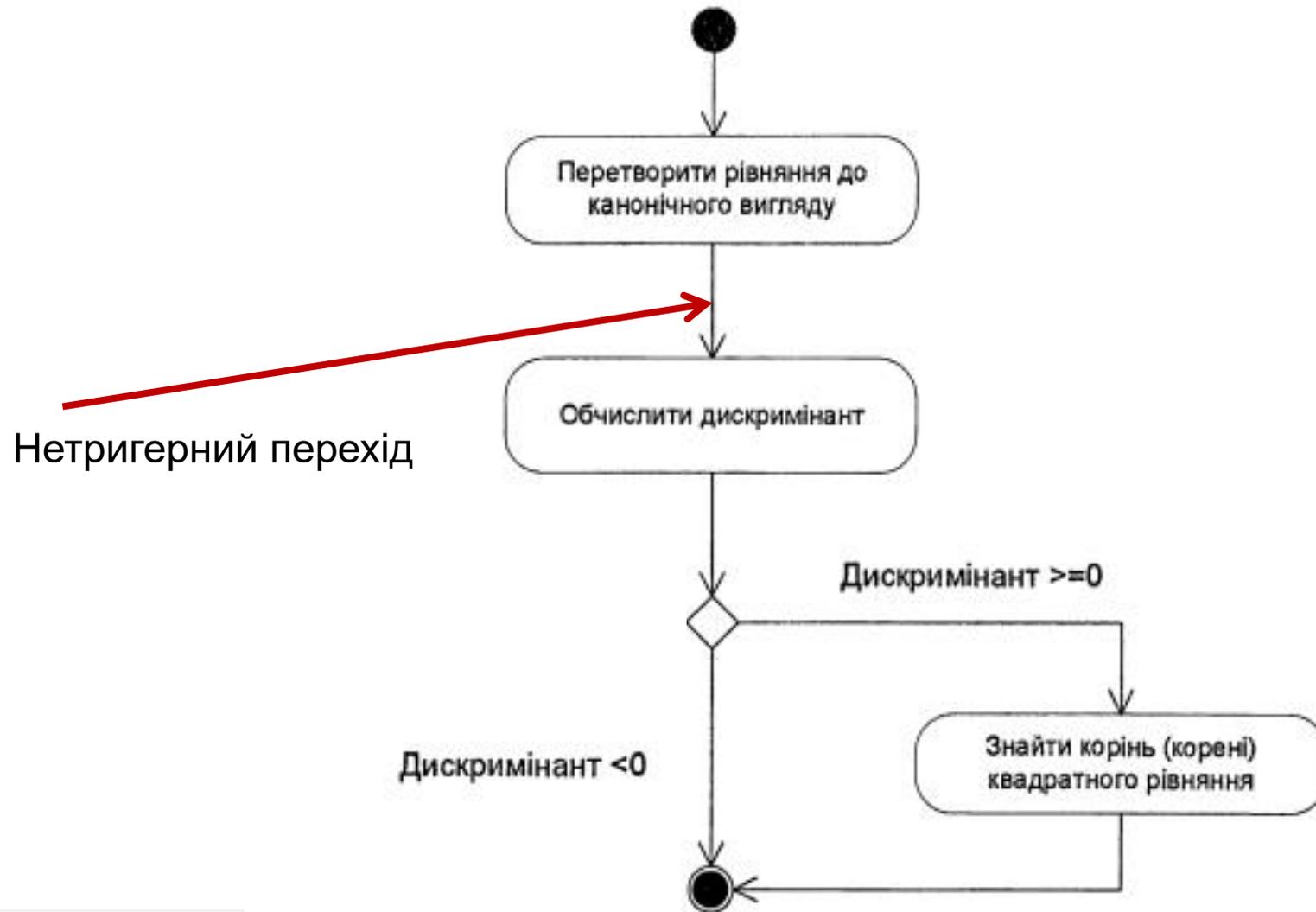


- Кожна діаграма діяльності повинна мати єдиний початковий і єдиний кінцевий стани.
- При цьому кожна діяльність починається в початковому стані і закінчується в кінцевому стані.
- Саму діаграму діяльності прийнято розташовувати так, щоб дії слідували зверху вниз. У цьому випадку початковий стан зображається у верхній частині діаграми, а кінцевий - в її нижній частині.

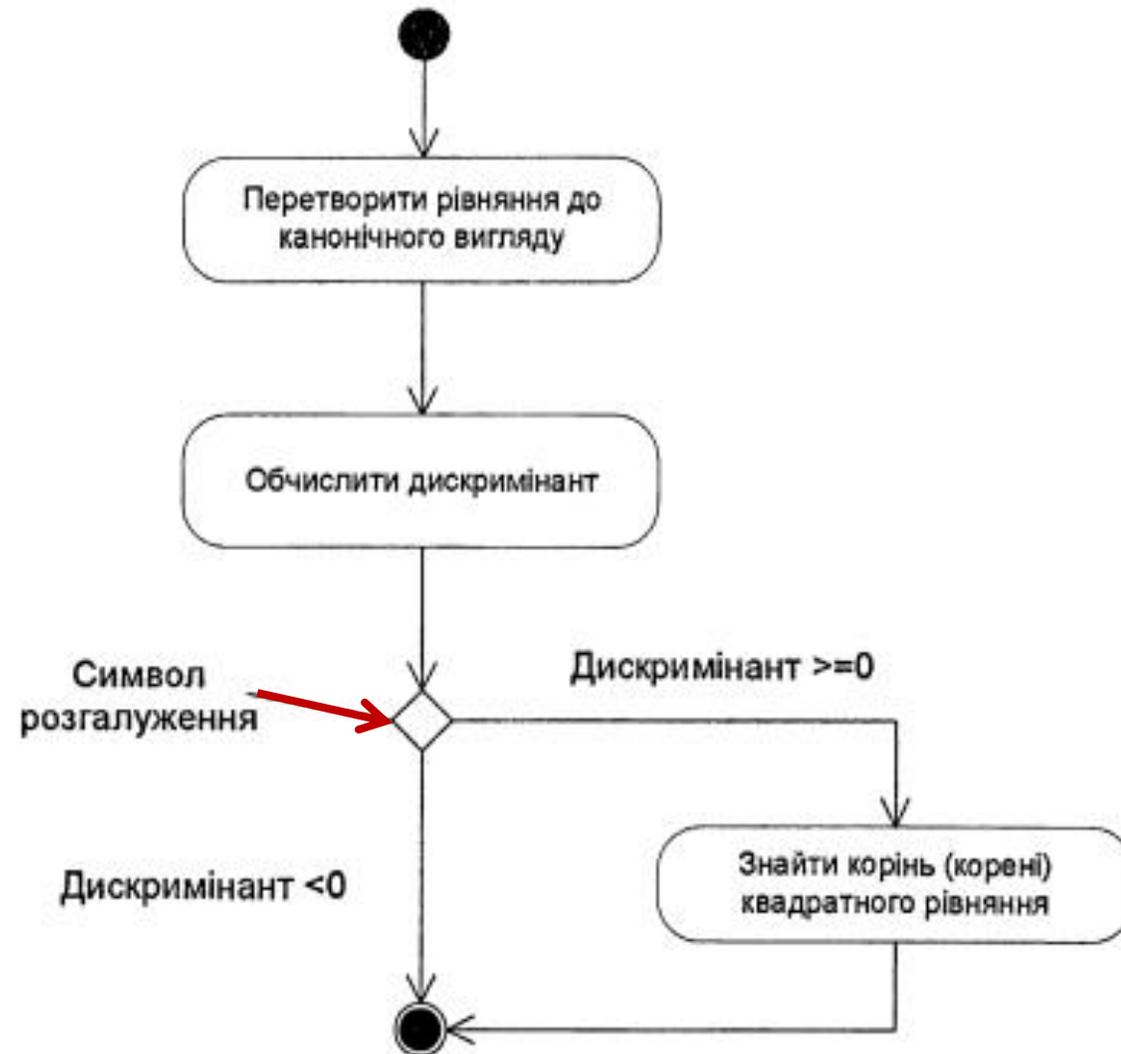
Стан дії (action state)



Переходи



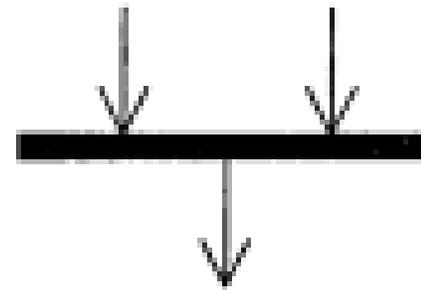
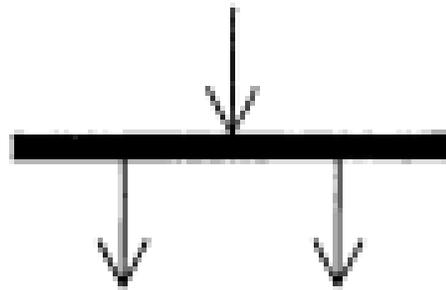
Розгалуження



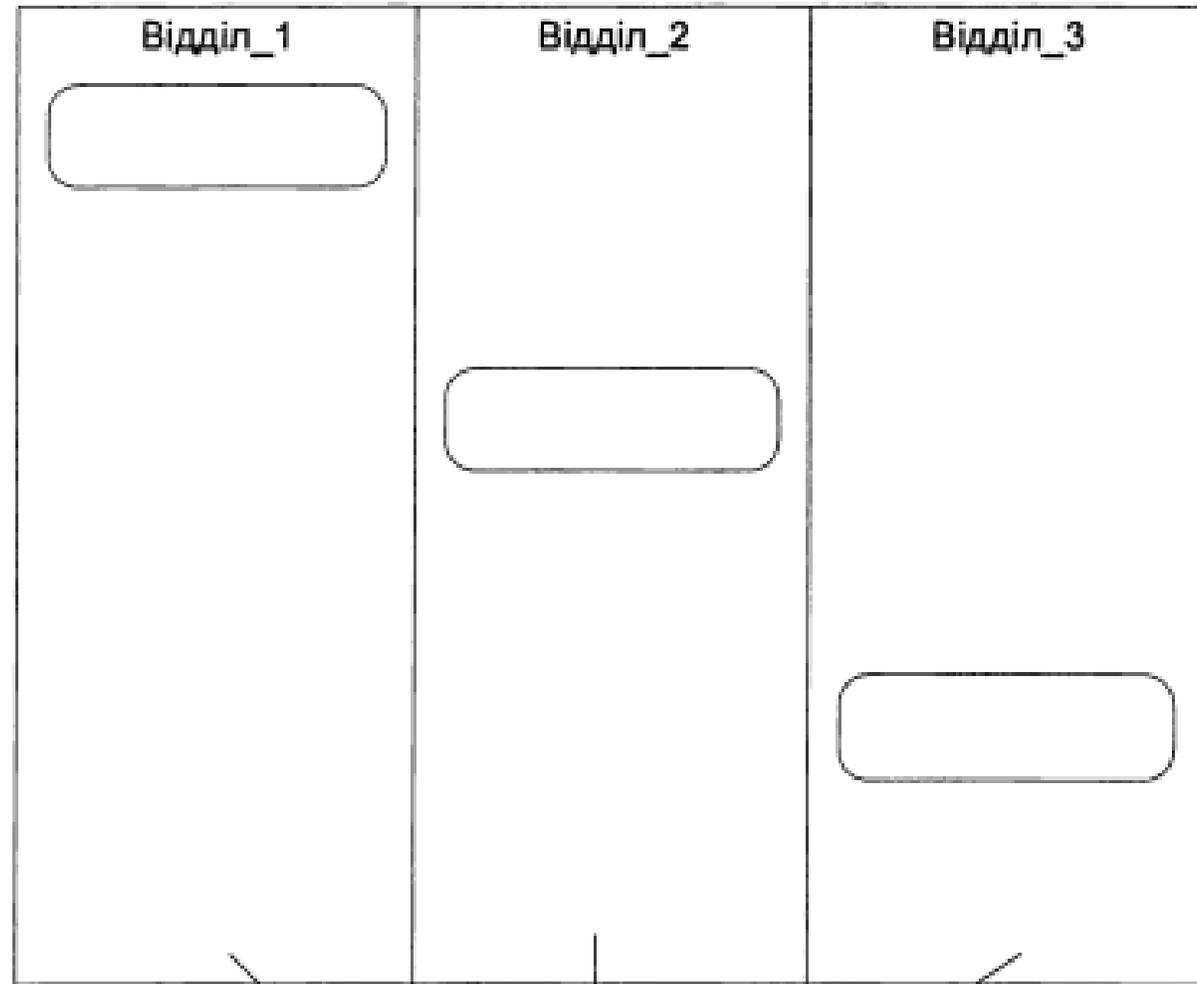
Паралельні гілки

- Один з найбільш значущих недоліків звичайних блок-схем алгоритмів пов'язаний із проблемою зображення паралельних гілок окремих обчислень. Оскільки розпаралелювання обчислень істотно підвищує загальну швидкодію програмних систем, необхідні графічні примітиви для зображення паралельних процесів. У мові UML з цією метою використовується спеціальний символ для розділення і злиття паралельних обчислень або потоків керування.

Паралельні гілки



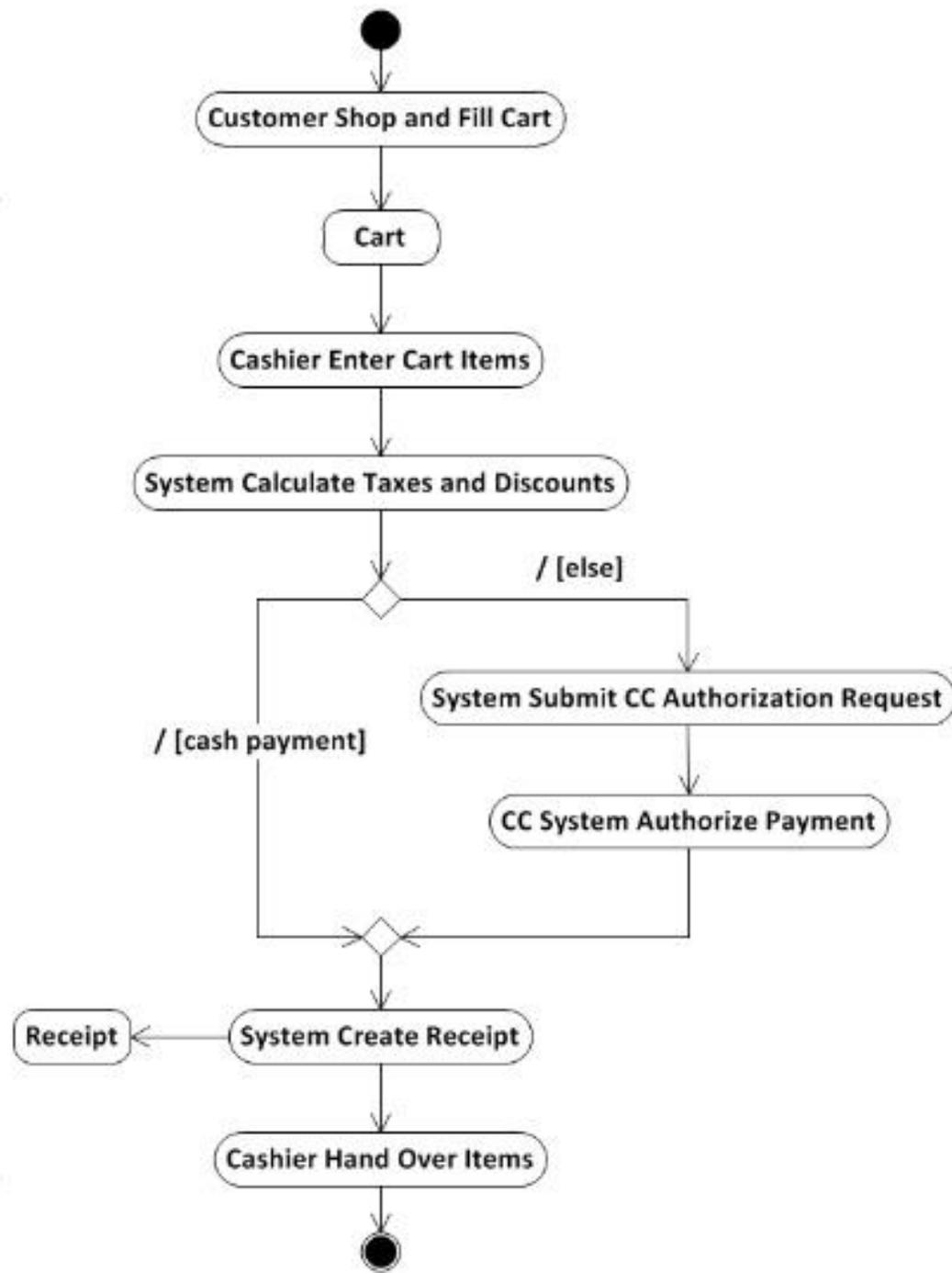
Доріжки

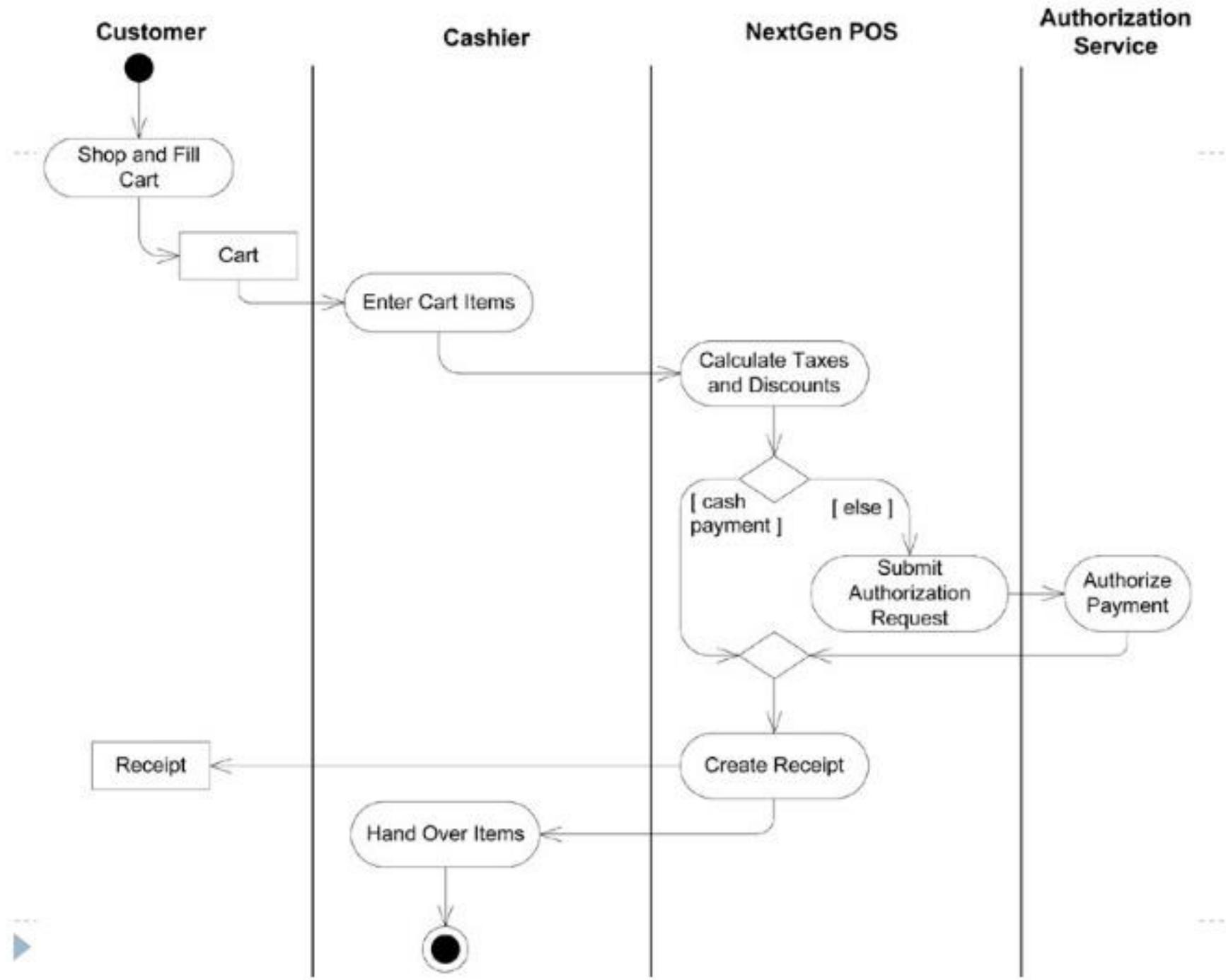


Доріжки на діаграмі діяльності

- Дії на діаграмі діяльності виконуються над тими чи іншими об'єктами. Ці об'єкти або ініціюють виконання дій, або визначають деякий результат цих дій.
- При цьому дії специфікують виклики, які передаються від одного об'єкту графу діяльності до іншого. Оскільки в такому ракурсі об'єкти відіграють певну роль в розумінні процесу діяльності, іноді виникає необхідність явно вказати їх на діаграмі діяльності.

Об'єкти на діаграмі діяльності





- 1. Визначити вимоги до системи
- 2. Скласти список акторів
- 3. Скласти список (таблицю) варіантів використання)
- 4. Побудувати діаграму варіантів використання
- 5. Для кожного варіанту використання побудувати діаграму діяльності
- 6. Побудувати діаграми станів для кожного об'єкта та послідовності



UseCase модель

Послідовність побудови



Запитання?

