

**Об'єтно-орієнтований
підхід до аналізу**

**Розробка UML діаграми
варіантів використання
системи**

Система моделей

Моделі

Структурний
підхід

Об'єктна

Функціональна

Управління

Організаційна

Технічна

Оціночний
підхід

Час розв'язування

Вартість опрацювання даних

Надійність процесів

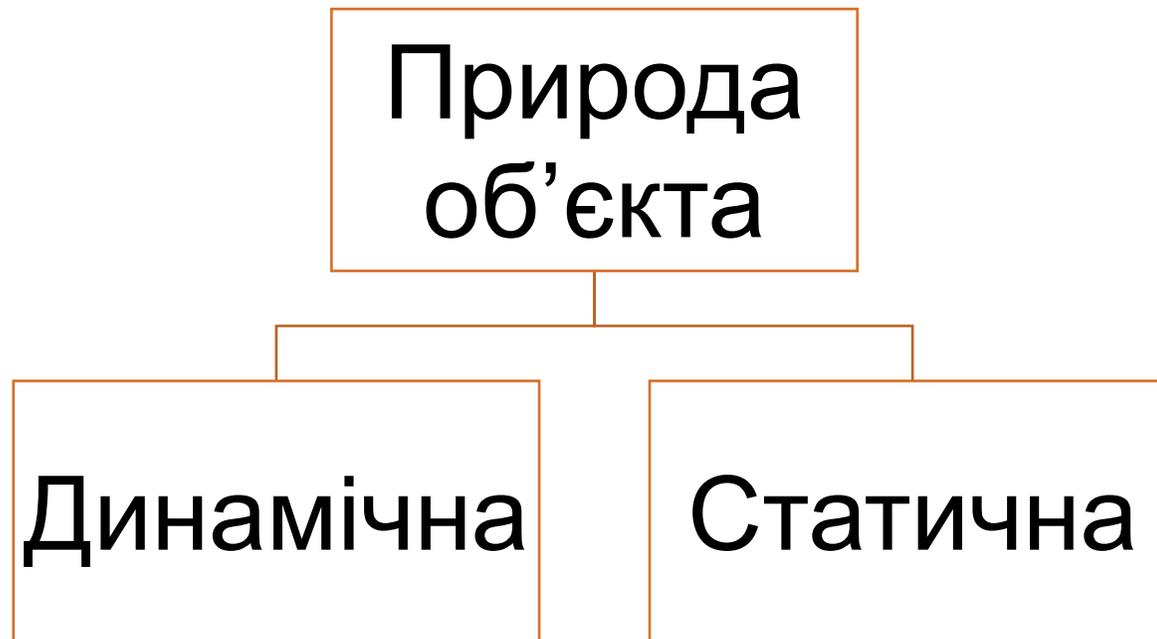
Показники ефективності

Об'єктна модель

відображає склад матеріальних та інформаційних об'єктів предметної області



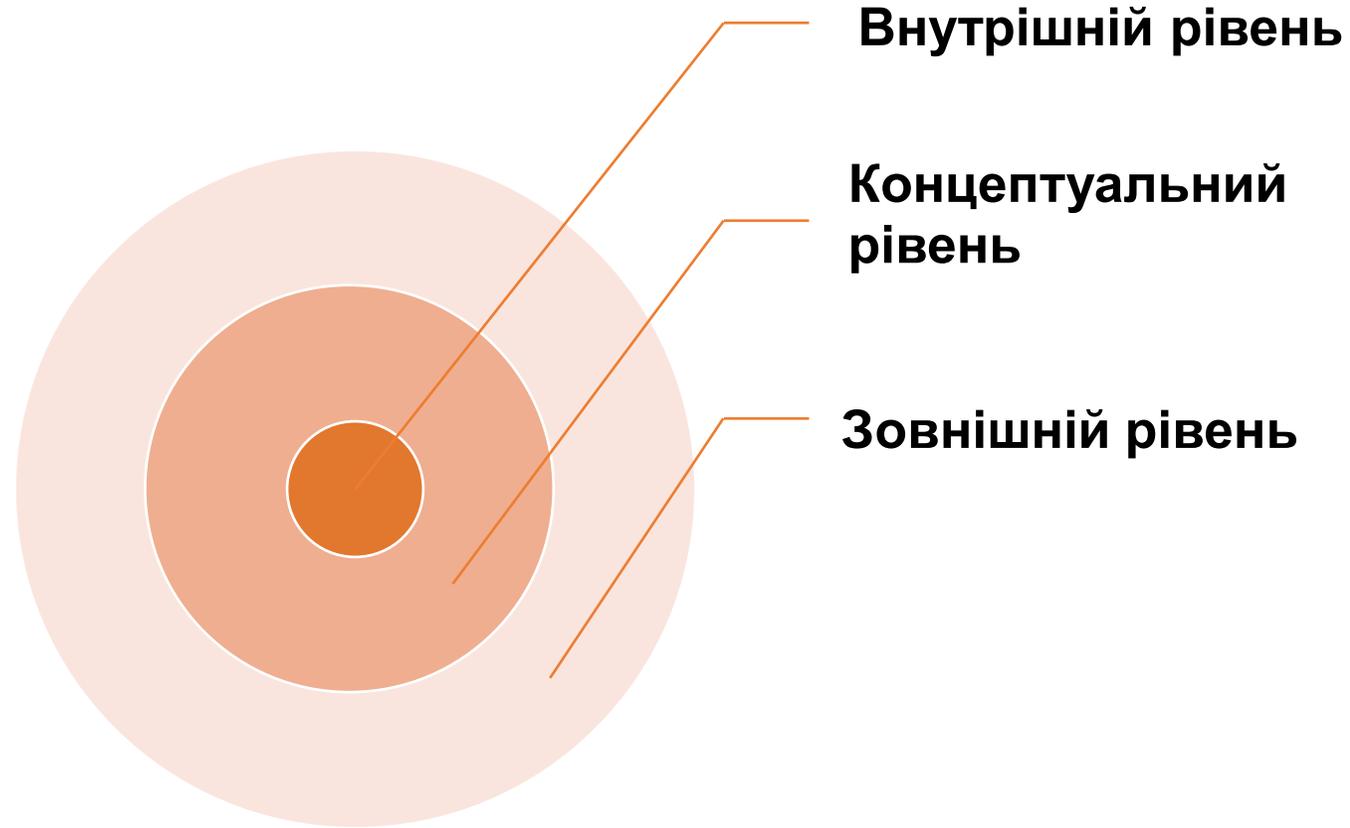
Об'єкт – це сутність, яка використовується при виконанні певної функції або операції (перетворення, опрацювання, формування і т.д.)



Об'єктна структура

Деталізація моделі

- ✓ **Зовнішній** – виділяються основні види матеріальних об'єктів та основні види інформаційних об'єктів і-або документів;
- ✓ **Концептуальний** – уточнення складу класів об'єктів, визначення їх атрибутів, взаємозв'язків між ними;
- ✓ **Внутрішній** – відображення файлів бази даних, вхідних та вихідних документів ІС. Модель бази даних відображає зберігання умовно-постійної інформації та накопичуваної змінної інформації



Основні поняття ООА

-  **Об'єкт** – предмет або явище, яке має чітку поведінку, володіє певним станом та індивідуальністю
-  **Клас** – множина об'єктів, пов'язаних спільністю структури і поведінки

Мова моделювання UML

UML (Уніфікована мова моделювання) - мова для моделювання програмних рішень, структур додатків, поведінки системи та бізнес-процесів. Включає **14 типів діаграм**, які дозволяють змодельовувати поведінку системи.

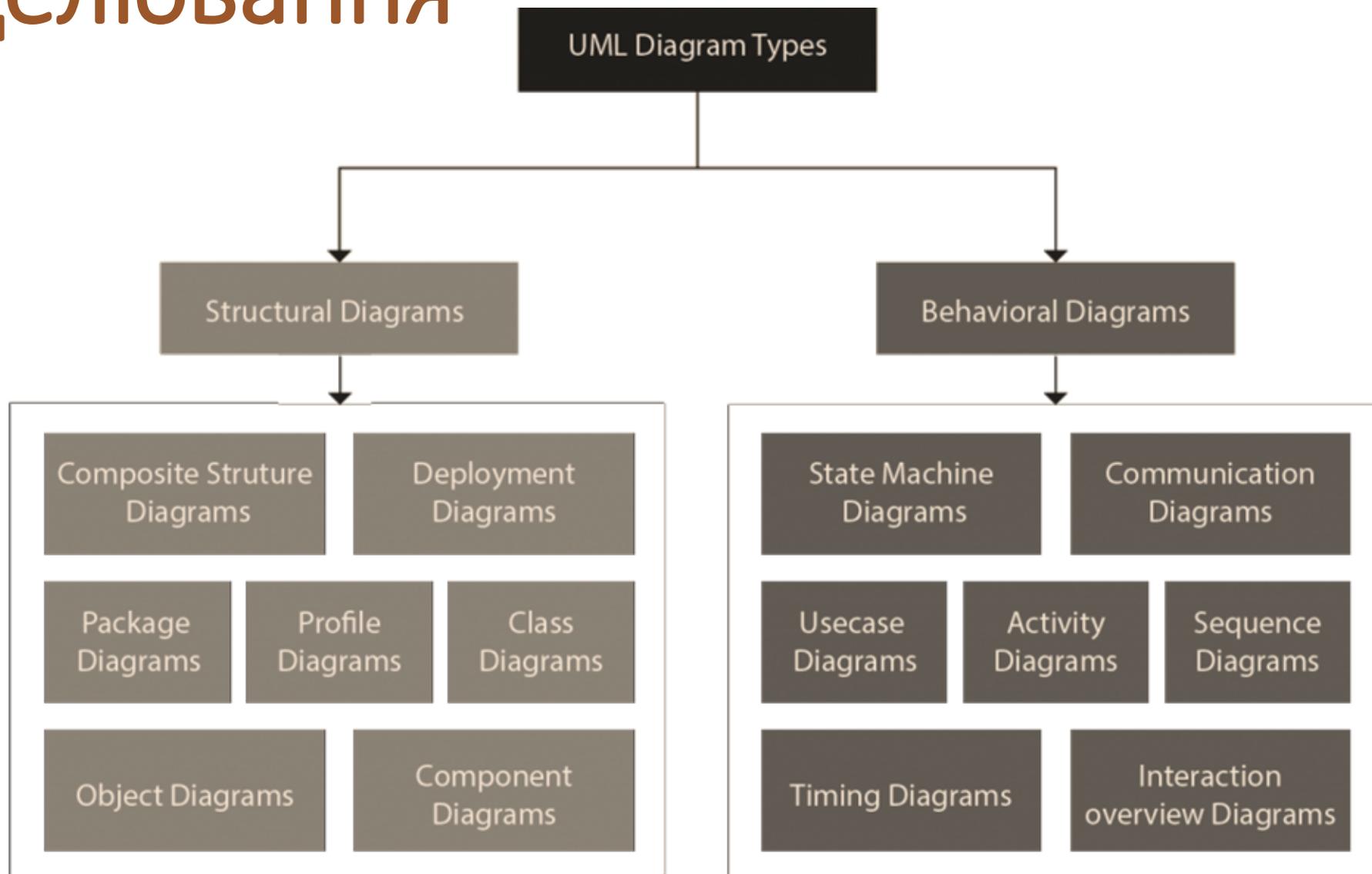


Структурні моделі - показують речі в змодельованій системі



Поведінкові моделі - описують, як об'єкти взаємодіють один з одним для створення функціонуючої системи

Мова моделювання UML



UseCase модель – діаграма варіантів використання



це тип поведінкової діаграми UML , який часто використовується для аналізу різних систем;



дозволяють візуалізувати різні типи ролей у системі та те, як ці ролі взаємодіють із системою.

Основні елементи:



Актор



Система



Варіант
використання



Пакет

Use Case модель: Етапи моделювання

- ✓ Визначити вимоги до системи
- ✓ Ідентифікація акторів
- ✓ Скласти список (таблицю) варіантів використання
- ✓ Побудувати діаграму варіантів використання
- ✓ Для кожного варіанту використання побудувати діаграму діяльності
- ✓ Побудувати діаграми станів для кожного об'єкта та послідовності

Визначення вимог до системи

Вимога (requirement) – це опис необхідних або бажаних властивостей продукту.

- ✓ Загальне формулювання задачі
- ✓ Цілі
- ✓ Функції системи
- ✓ Атрибути системи

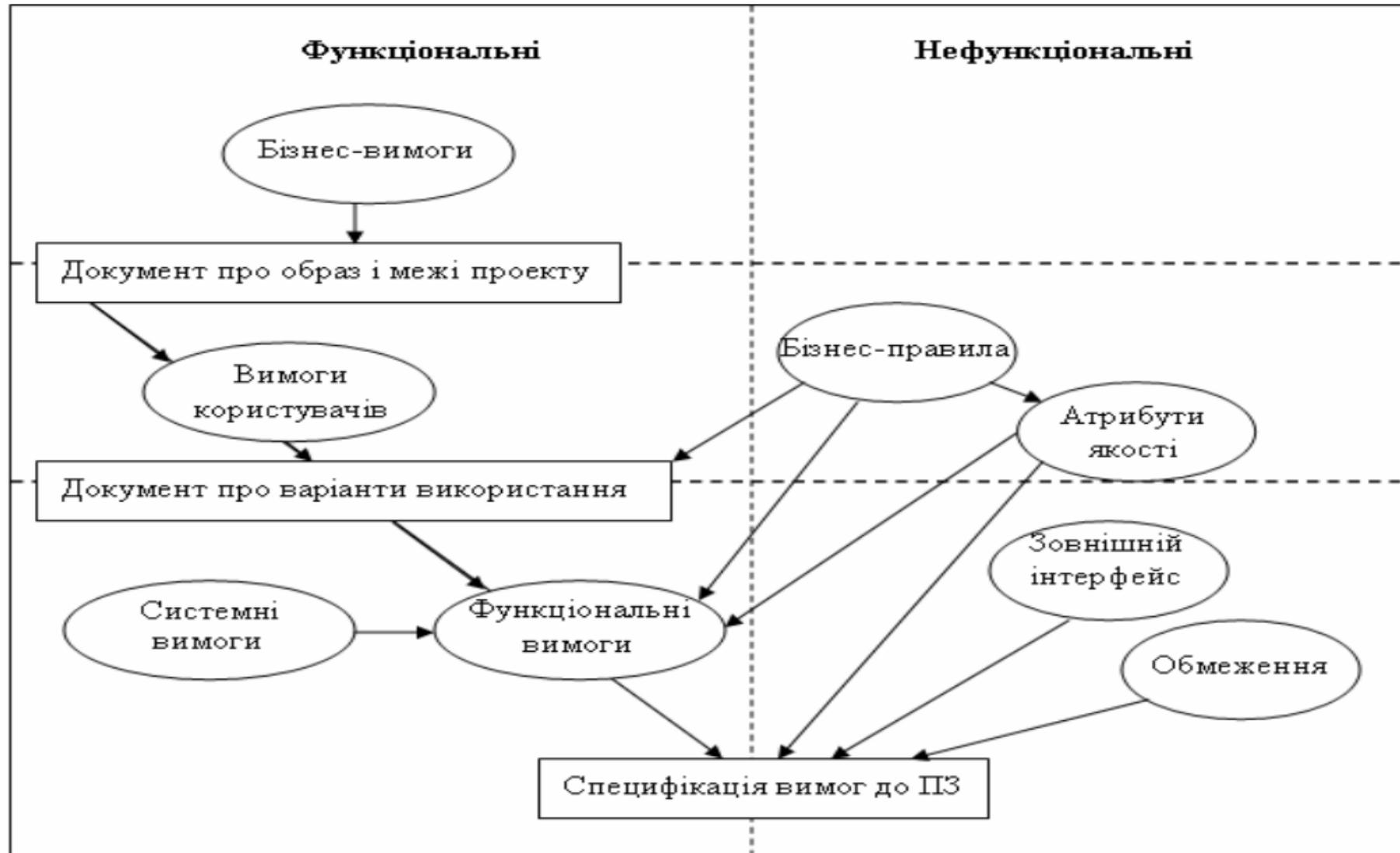
Бізнес- вимоги

Вимоги користувачів

Функціональні вимоги

Нефункціональні вимоги

Специфікація вимог



Характеристика ВИМОГ

Повнота

Коректність

Здійсненність

Необхідність

Пріоритетність

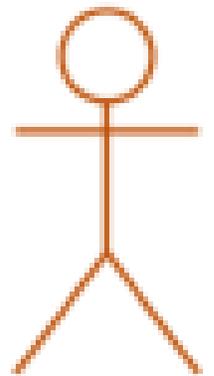
Недвозначність

Можливість перевірки

Ідентифікація акторів

- ✓ **Актор UML** (англ. Unified Modeling Language) — це роль, що виконується сутностями (людина, інша система тощо), які взаємодіють з системою.
- ✓ **Актори** є зовнішніми об'єктами по відношенню до системи.

Student є зовнішнім об'єктом по відношенню до системи управління факультетом, але в системі зберігатимуться дані про студентів у вигляді об'єктів класу StudentDetails



Actor

Ідентифікація акторів

Щоб визначити акторів, потрібно відповісти на такі питання:
«Хто або що використовує систему або взаємодіє з нею і які ролі вони виконують при взаємодії з нею?»

-  Що налаштовує систему?
-  Хто запускає та вимикає систему?
-  Хто обслуговує систему?
-  Які системи взаємодіють з нею?
-  Хто або що отримує або постачає інформацію системі?
-  Чи відбувається щось в точно визначений час?

Ідентифікація прецедентів



Прецедент — варіант (випадок) використання системи одним із акторів.

Банківська система



Відкрити
рахунок

Перевірити
баланс

Здійснювати
платежі

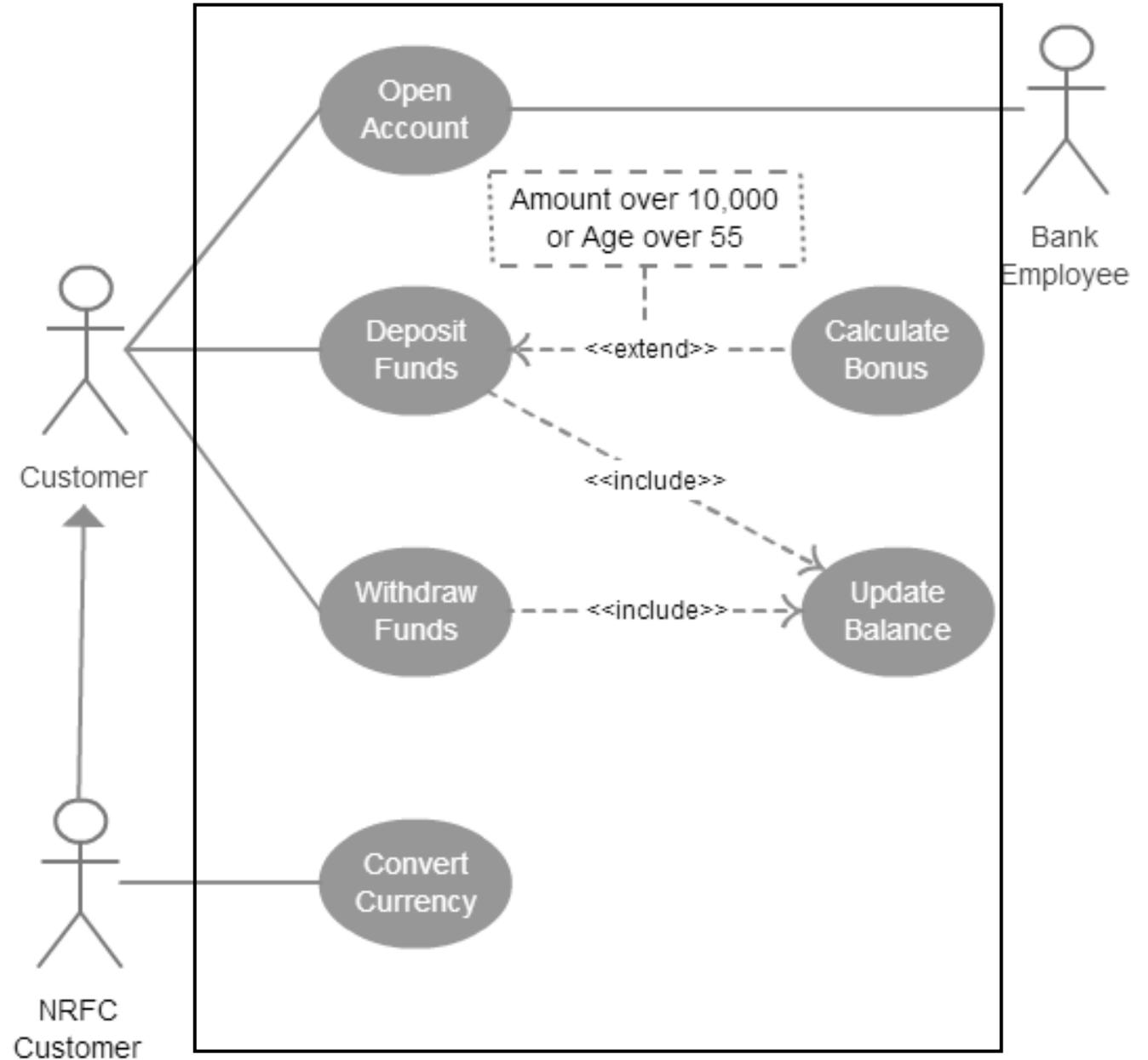
Вносити кошти

Use Case

Випадки використання верхнього рівня завжди повинні забезпечувати повну функцію, необхідну актору. Залежно від складності системи, ви можете **розширити (extend)** або **включити (include)** випадки використання

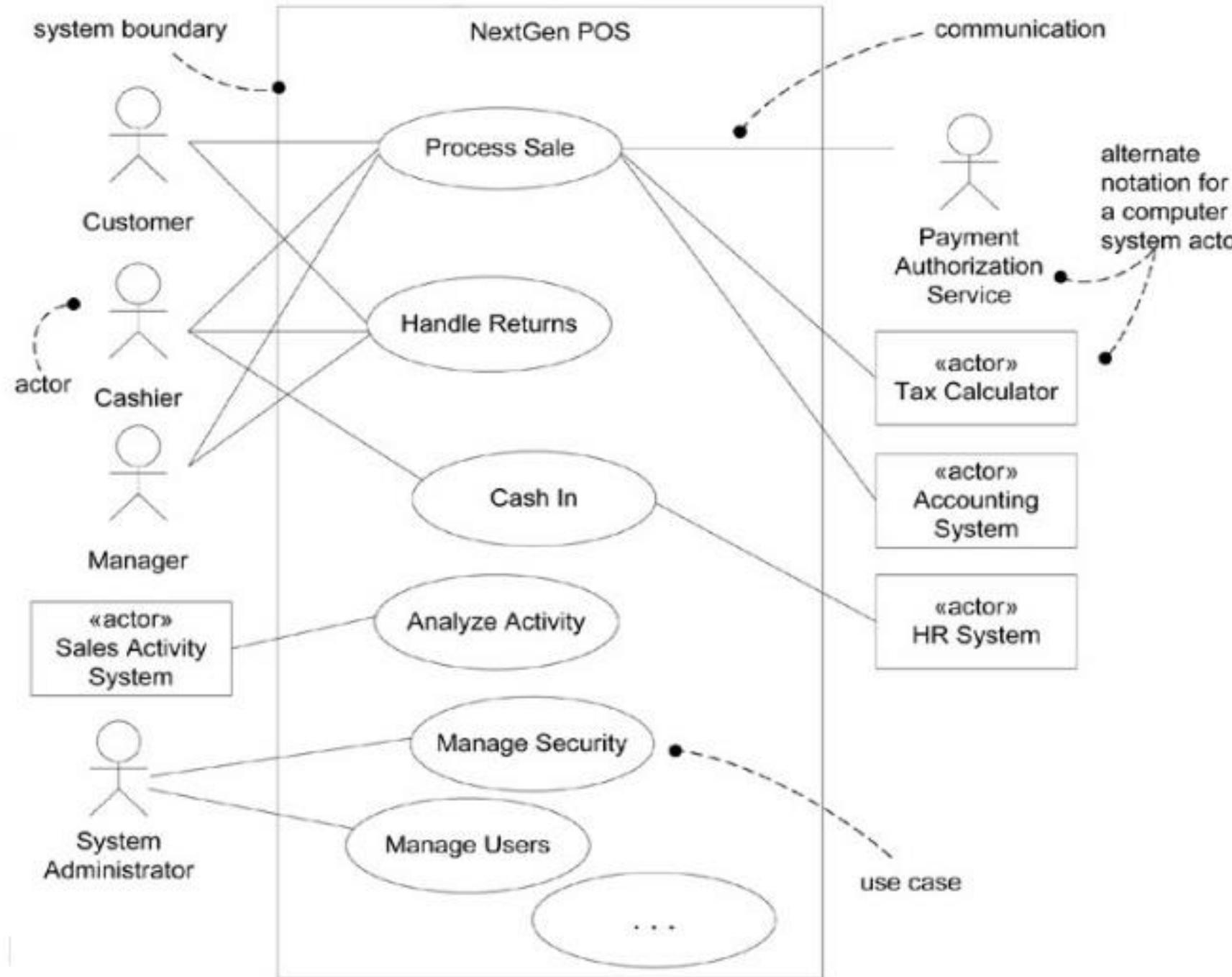
Приклад моделі

Банківська система



Приклад моделі

Система продажу



Сценарій виконання прецеденту

Приклад 1. Розблокувати обліковий запис користувача (простий короткий приклад без альтернативного потоку подій):

Діючі ролі (Актори)	Адміністратор, Система
Ціль	Змінити статус облікового запису користувача на активний.
Передумова	Обліковий запис користувача не активний.
Успішний сценарій:	<p>1.Адміністратор вибирає користувача та активує «Розблокувати».</p> <p>2.Система переключає обліковий запис користувача у статус «активний», і надсилає повідомлення (<i>тут можна послатися на текст повідомлення зі списку повідомлень, див. примітку нижче</i>) користувачеві на email (якщо User Account → email не порожньо).</p>
Результат	Обліковий запис користувача був переведений у статус активний.

Сценарій виконання прецеденту

Приклад 2. Авторизація користувача

Діючі лиця	Користувач, Система
Цілі	Користувач: авторизуватися в системі та почати працювати; Система: ідентифікувати користувача та його права.
Успішний сценарій: 1. Користувач запускає систему. Система відкриває сесію користувача, пропонує ввести логін та пароль. 2. Користувач вводить логін та пароль. 3. Система перевіряє логін та пароль. 4. Система створює запис в історії авторизацій (IP-адреса користувача, логін, дата) 5. Система видає користувачеві повідомлення з приводу успішної авторизації	
Результат	Користувач успішно авторизований і може працювати із системою.
Розширення (альтернативні сценарії):	
*а	Нема доступу до БД. Система видає повідомлення (<i>посилання повідомлення</i>). Результат: користувач не може увійти.
1а	У налаштуваннях безпеки для цієї IP адреси існує заборона входу в систему. Результат: форма логіну не надається, система видає повідомлення користувачу
2а	Користувач вибирає: "Нагадати пароль". Викликається сценарій «нагадати пароль».
3а	Користувач із введеними логіном та паролем не знайдено. Результат: відмова у авторизації. Система видає повідомлення. Перехід крок 2.
3б	Кількість невдалих спроб авторизуватися досягла максимального, встановленого у налаштуваннях. Результат: користувач не може увійти. Видається повідомлення. Вхід з IP-адреси Користувача заблоковано на час, встановлений у налаштуваннях.