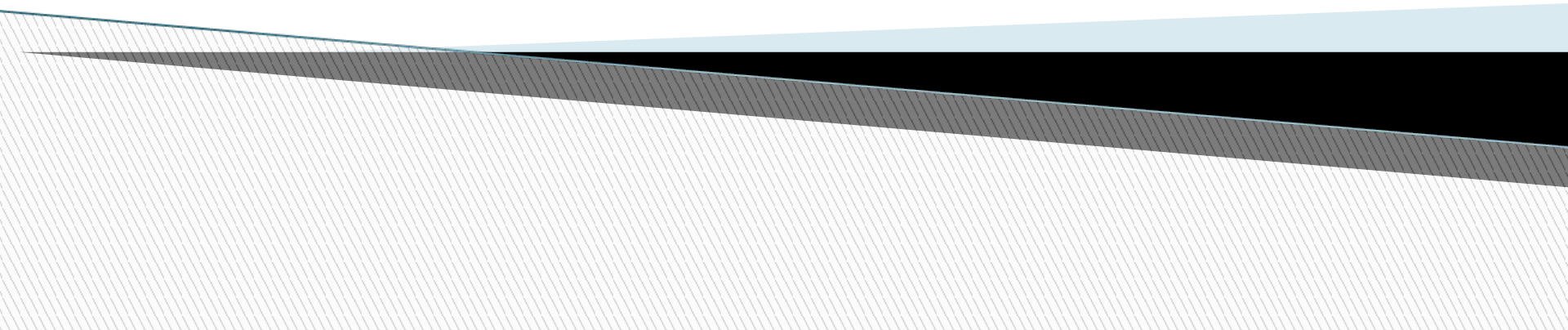


# Інженерія вимог до програмного забезпечення

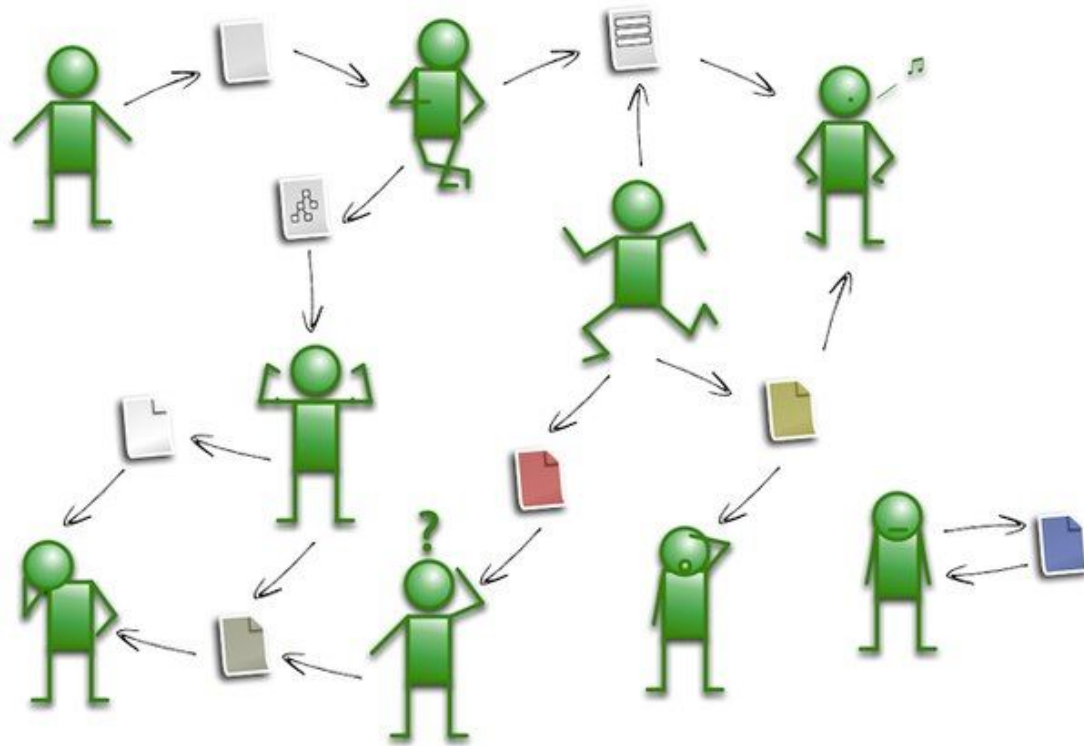
*Лекція 2.*

*Ч.1*

*Розділ 2. Тема 2.1*



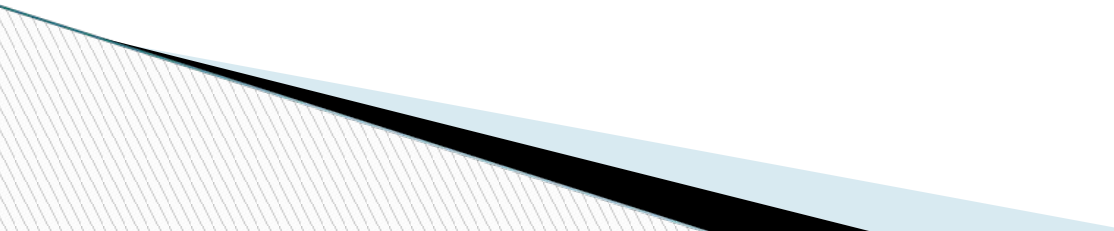
# *Загальні відомості про інженерію вимог*



**Галузь знань "Вимоги до ПЗ  
(Software Requirements)"  
складається з наступних розділів:**



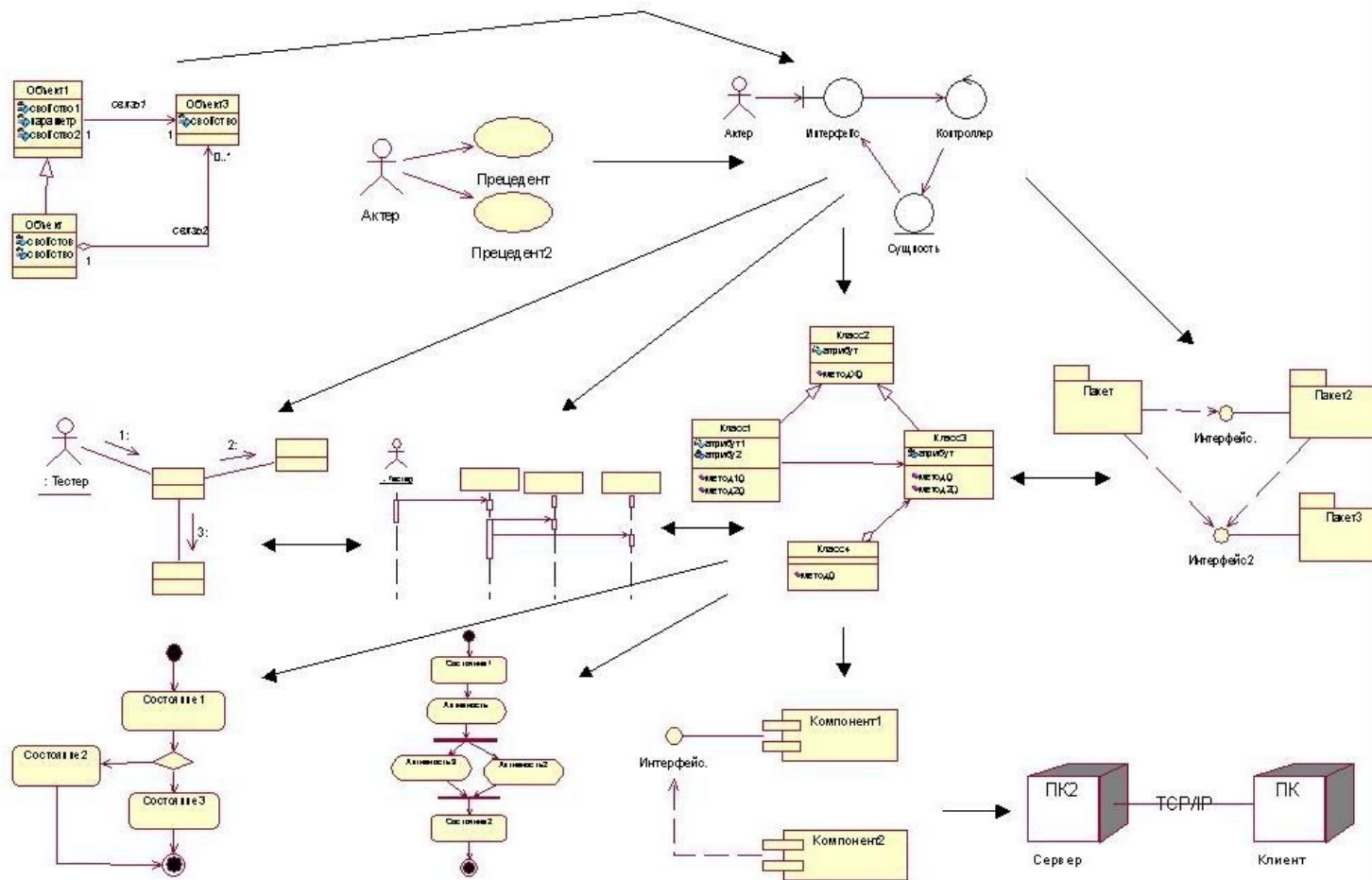
## *Аналіз вимог до ПЗ - це ...*

- ✓ процес збору вимог до програмного забезпечення,
  - ✓ систематизації вимог,
  - ✓ документування вимог,
  - ✓ аналіз, виявлення суперечностей, неповноти вимог,
  - ✓ вирішення конфліктів у процесі розробки програмного забезпечення.
- 

# *Методи інженерії вимог*

- Метод інженерії вимог С. Шлеєр та С. Меллора
- Метод інженерії вимог І. Джекобсона

# Метод інженерії вимог С. Шлеєр та С. Меллора



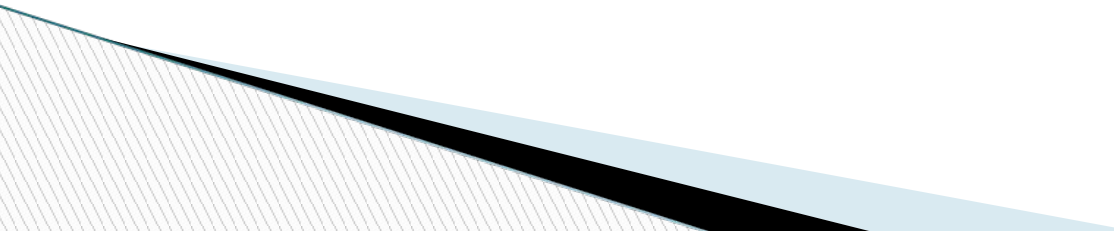
# Поняття «Вимога до ПЗ»

- ▣ **SWEBOK** (*Software Engineering Body of Knowledge*) Програмні вимоги (Software Requirements) – властивості програмного забезпечення, які повинні бути належним чином представлені в ньому для вирішення конкретних практичних завдань.
- ▣ **RUP** (*Rational Unified Process*) Вимога – це умова або можливість, якій повинна відповідати система.
- ▣ **IEEE** (*I triple E, Institute of Electrical and Electronics Engineers, Інститут інженерів з електротехніки та електроніки*) **Standard Glossary of Software Engineering Terminology (1990)**
  1. Умови або можливості, необхідні користувачу для вирішення проблем або досягнення цілей;
  2. умови або можливості, якими має володіти система або системні компоненти, щоб виконати контракт або задовольняти стандартам, специфікаціям або іншим формальним документам;
  3. документоване уявлення умов або можливостей для пунктів 1 та 2.
- ▣ **Дорфман (Dorfman), Тэйер (Thayer)** (*Леффінгуел. Принципи роботи з вимогами ПЗ*)
  1. Якась властивість програмного забезпечення, необхідна користувачеві для вирішення проблеми при досягненні поставленої мети.
  2. Якась властивість програмного забезпечення, яким повинна володіти система або її компонент, щоб задовольняти умовам контракту, стандарту, специфікації або іншій формальній документації.
- ▣ **Ian Sommerwille & Pete Sawyer** Вимога – це специфікація того, що повинно бути отримано. Вимоги описують поведінку системи або атрибути та властивості системи. Вимоги можуть бути і обмеженнями на процес розробки системи.

*Вимоги* - це властивості, якими має володіти ПЗ для адекватного визначення функцій, умов і обмежень виконання ПЗ, а також обсягів даних, технічного забезпечення та середовища функціонування.

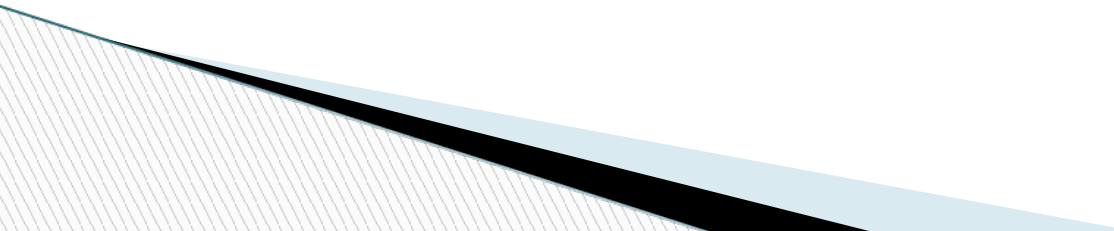


## *Властивості вимог:*

- Ясність, недвозначність;
  - повнота і несуперечність;
  - необхідний рівень деталізації;
  - простежуваність;
  - тестованість і перевірюваність;
  - модифікованість.
- 

# Класифікація вимог

**SWEBOK** - не описує підходи до класифікації вимог, а описує можливості угруповання вимог відповідно до їх характеристик.

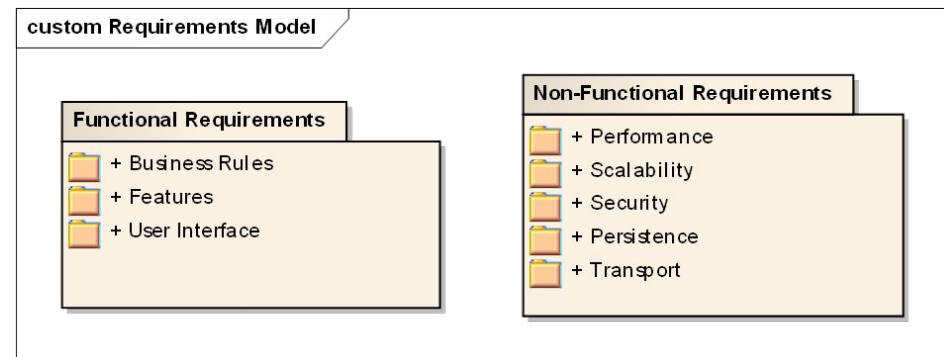
- Вимоги до продукту та процесу
  - Функціональні та нефункціональні вимоги
  - Незалежні властивості
  - Вимоги з кількісною оцінкою
  - Системні вимоги та програмні вимоги
- 

# *Класифікація вимог*

*Вимоги до продукту і до процесу* визначають умови функціонування і режими роботи ПЗ в операційному середовищі, обмеження на структуру і пам'ять комп'ютерів, на принципи взаємодії програм і комп'ютерів та інше.

Найбільш часто модель вимог поділяють на дві моделі:

- *модель функціональних вимог* - містить вимоги і властивості, що визначають очікувану функціональну поведінку системи;
- *модель нефункціональних вимог* - містить обмеження на рівні продуктивності, очікуваної від системи (наприклад, час відгуку, стійкість шифрування, число транзакцій за секунду).



# *Класи нефункціональних вимог:*

- вимоги конфідесійності;
- відмовостійкість;
- число клієнтів, які мають одночасно доступ до системи;
- вимоги безпеки;
- час очікування відповіді на звернення до системи;
- виконавські якості системи (обмеження щодо ресурсів пам'яті, швидкість реакції на звернення до системи та інше).

# ***Класифікація вимог***

***Вимоги з кількісною оцінкою*** визначають показники якості ПП.

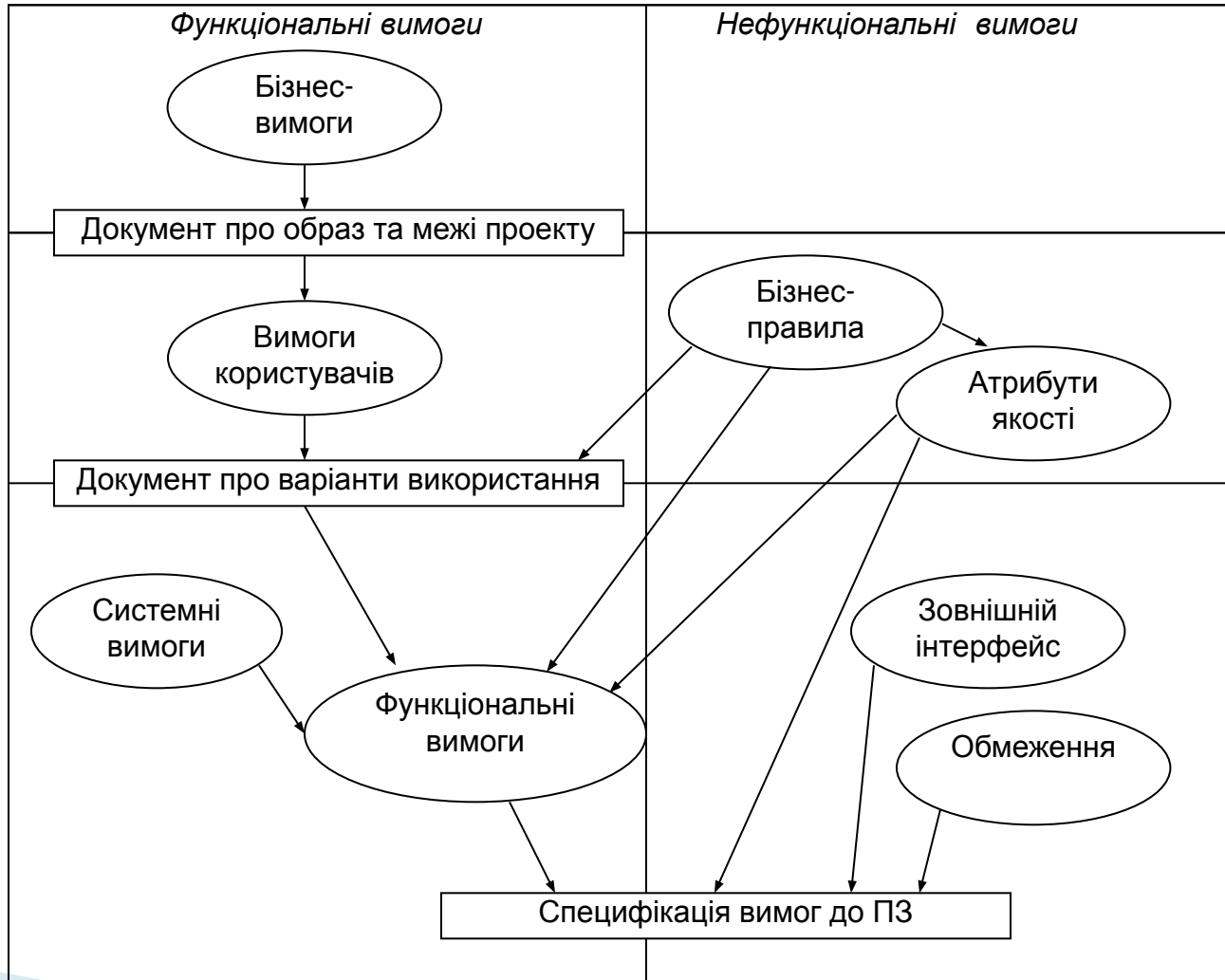
***Показник якості (продукції)*** - це кількісна характеристика одного або декількох властивостей продукції, що входять до її якості, то розглядається відповідно до певних умов її створення та експлуатації або споживання.

# *Класифікація вимог*

**Програмні вимоги** - Software Requirements - властивості програмного забезпечення, які повинні бути належним чином представлені в ньому для вирішення конкретних практичних задач.

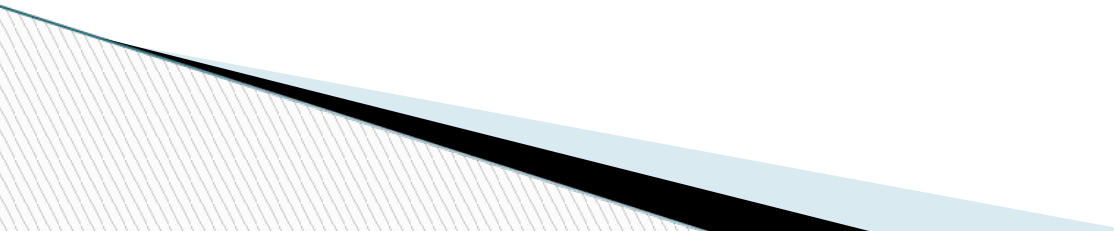
# Класифікація вимог

К.ВІГЕРС





# *Яких вимог не повинно бути!*

- Деталей дизайну або реалізації
  - Даних про планування проекту
  - Відомостей про тестування
- 

# *Класифікація вимог*

Д. Леффінгуел. Піраміда вимог



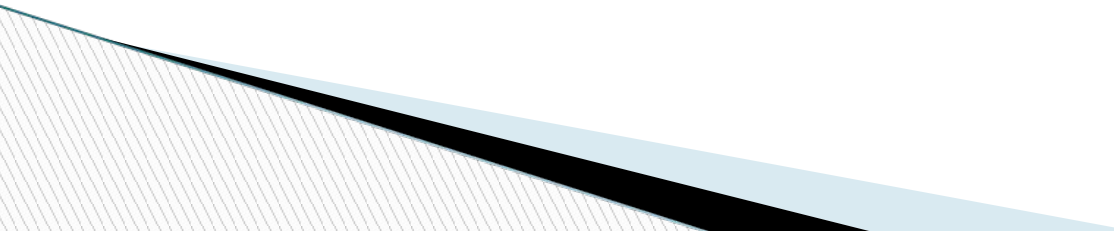
# Зацікавлені особи

Ніде більше, як на стадії збору вимог, так тісно не пов'язані інтереси всіх зацікавлених у проекті осіб з успіхом проекту.

До зацікавлених у проекті осіб належать:

1. **Замовники**, які фінансують проект або набувають продукт для вирішення якихось бізнес-завдань.
2. **Користувачі**, які взаємодіють безпосередньо чи не безпосередньо з застосуванням (підклас замовників).
3. **Аналітики вимог**, які пишуть вимоги і передають їх розробникам.
4. **Розробники**, які створюють, розгортають і підтримують продукт.
5. **Тестери**, які визначають відповідність поведінки ПО бажаного.
6. **Технічні письменники**, які відповідають за створення керівництва користувачів, тренувальні матеріали та довідкову систему.
7. **Менеджер по проекту**, який планує процес і керує командою розробників аж до успішного випуску продукту.
8. **Співробітники правового відділу**, які стежать, щоб продукт не суперечив усім чинним законам і постановам.
9. **Розробники**, які повинні побудувати продукт, що містить дане ПЗ.
10. **Співробітники відділу продажів і маркетингу**, виїзної служби підтримки та інші, кому доведеться працювати з продуктом і його користувачами.

# *Ідентифікація зацікавленої особи*

- кожен, хто користується системою (користувачі та обслуговуючий персонал);
  - будь-який, хто отримує вигоду з системи (функціональну, політичну, фінансову і соціальну);
  - залучений в покупку або забезпечення системи організації, які регулюють аспекти системи (фінансові, безпека, та інші)
  - люди або організації, налаштовані проти системи (негативні зацікавлені особи),
  - організації, відповідальні за систему, які взаємодіють з системою згідно з проектом
  - ті організації, які об'єднуються горизонтально з організацією, для якої аналітик проектує систему
- 

# Особливості збору та аналізу бізнес-вимог

## Продукт під замовлення

### Характеристики процесу:

- сектор ринку з самого початку може бути не визначено;
- мети продукту ґрунтуються на конкурентному аналізі.

### Особливості збору та аналізу бізнес-вимог:

- відразу за визначенням вихідних передумов (стимулів) йде огляд конкурентів;
- далі йде визначення цільового сегмента ринку і потреб його замовників;
- в кінці визначаються цілі продукту і критерії його успіху.

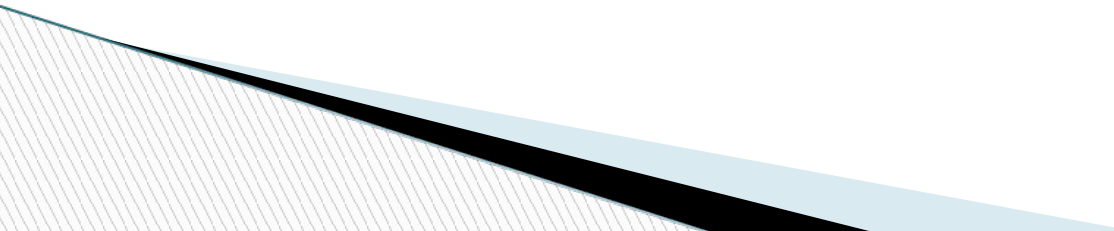
# Особливості збору та аналізу бізнес-вимог

## Встроєні застосунки

### Характеристики процесу:

- залежать від того, чи є це продуктом під замовлення (наприклад, ПЗ для мікрохвильової печі) або продуктом для відкритого ринку (наприклад, ПЗ мобільних телефонів).

# *Визначення стимулів*

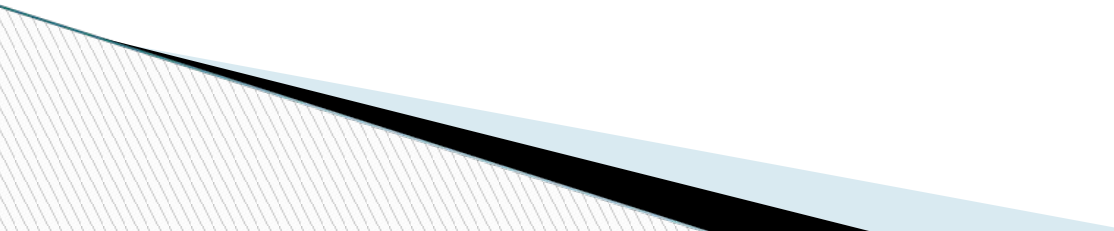
- потреба ринку;
  - виробнича необхідність;
  - потреба замовника;
  - технічний прогрес;
  - юридичні обмеження або норми.
- 

# *Визначення цілей продукту і критеріїв успіху*

## **фінансові:**

- ▣ Досягти обсягу продажів  $X$  одиниць або доходу, рівного  $\$ Y$ , за  $Z$  місяців.
- ▣ Отримати  $X\%$  прибутку або доходу з інвестицій протягом  $Y$  місяців.
- ▣ Заощадити  $X \$$  на рік, які зараз витрачаються на обслуговування системи.
- ▣ Зменшити витрати на підтримку на  $X\%$  за  $Z$  місяців.
- ▣ Отримати не більше  $X$  дзвінків у службу обслуговування по кожній одиниці товару і  $Y$  дзвінків по гарантії кожної одиниці товару протягом  $Z$  місяців після випуску товару.

## **нефінансові:**

- ▣ Досягти показника задоволення покупців, рівного, принаймі,  $X$ , протягом  $Y$  місяців з часу випуску товару.
  - ▣ Збільшити продуктивність обробки транзакцій на  $X\%$  і знизити рівень помилок даних до величини не більше  $Y\%$ .
  - ▣ Розробити надійну платформу для сім'ї пов'язаних продуктів.
- 

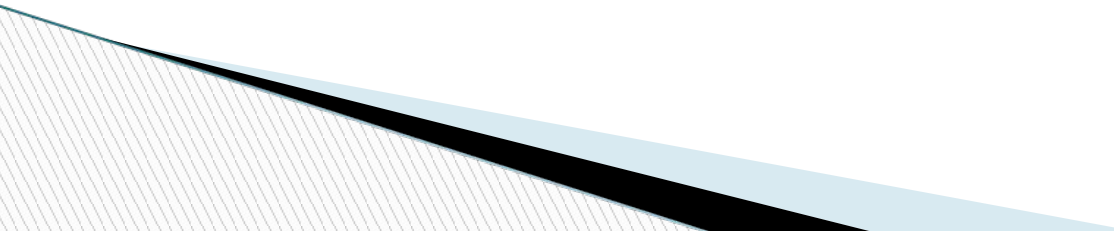


# ***Виявлення вимог***

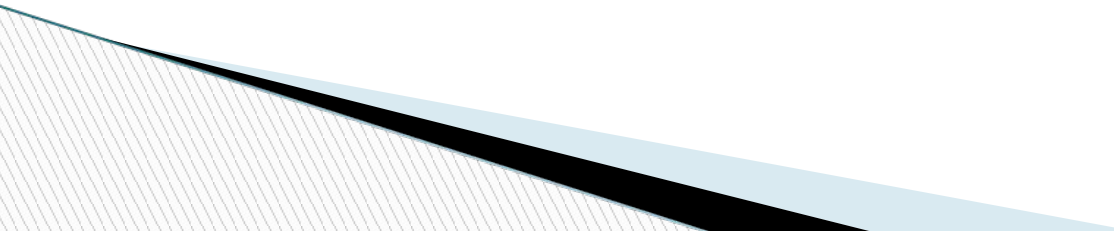
***Виявлення вимог – це ...***

***Модель процесу визначення вимог – це схема процесів ЖЦ, які ...***

# ***Методи виявлення вимог***

- Інтерв'ю, опитування, анкетування;
  - мозковий штурм, семінар;
  - спостереження за виробничою діяльністю, «фотографування» робочого дня;
  - аналіз нормативної документації;
  - аналіз моделей діяльності;
  - аналіз конкурентних продуктів;
  - аналіз статистики попередніх версій системи.
- 

# Проблеми при виконанні проектів виникають через:

- неформальний збір інформації, передбачуваної функціональності АС,
  - помилкових або неузгоджених нефункціональних вимог до системи, а також нерегламентованої процедури їх зміни,
  - та інше.
- 

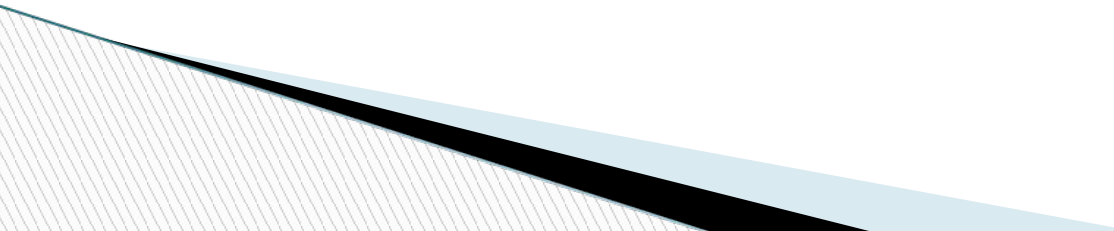
# Помилки та різночитання

Помилки і різночитання, які виникають при виявленні вимог до системи, виявляються одними з найдорожчих.

**Вимоги** - це те вихідне розуміння завдання розробниками, яке є основою всієї розробки.

*По-перше, щоб добре розібратися, якою має бути система наприклад автоматизації лікарні та система підтримки хімічних експериментів - треба попрацювати у відповідній області достатній час. Або якимось іншим способом **навчитися бачити проблеми даної предметної області зсередини.***

По-друге, позначається специфічність програмування як сфери діяльності. Для більшості користувачів і замовників вкрай **не просто сформулювати точне знання,** яке необхідно програмістам.



# Першочергові перешкоди

1. Образ і межі проекту ніколи не визначені ясно.
2. Замовники дуже зайняті, щоб працювати з аналітиками і програмістами над вимогами.
3. Заступники користувачів - менеджери по продуктах, по розробці, менеджери користувачів або маркетологи - викликаються говорити від імені клієнтів, але не точно представляють їхні потреби.
- 4 Вимоги існують тільки у головах "експертів", які працюють у вашій організації, і ніколи не фіксуються у письмовому вигляді.
5. Замовники наполягають, щоб критикувалися всі вимоги, без урахування їх пріоритетів.
6. Розробники отримують двозначну і неповну інформацію, тому при кодуванні їм доводиться робити припущення.
7. Взаємодія між розробниками і замовниками обмежується зовнішнім виглядом користувальницького інтерфейсу і не зачіпає того, що ж дійсно клієнти збираються робити за допомогою програми.
8. Ваші замовники підписують вимоги і потім постійно змінюють їх.
9. Проект розростається, коли ви приймаєте зміни вимог, графік при цьому порушується, тому що ніяких додаткових ресурсів не виділяється і ніякі функції не видаляються.
10. Запити на зміну вимог губляться по дорозі: ні ви, ні ваші клієнти не знають статус кожного запиту.
11. Замовники наполягають на певній функціональності, яку розробники і створюють, однак ці можливості системи клієнтам не потрібні.

***Помилки, допущені на стадії збору вимог, складають від 40 до 60% всіх дефектів проекту.***

