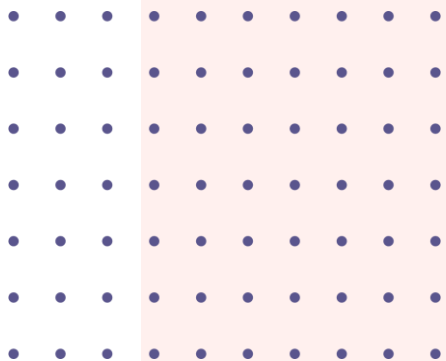


# Урок 1. Вступ до ІТ. Компанії та проєкти. Поняття тестування. Основні стандарти ISTQB

# Структура заняття

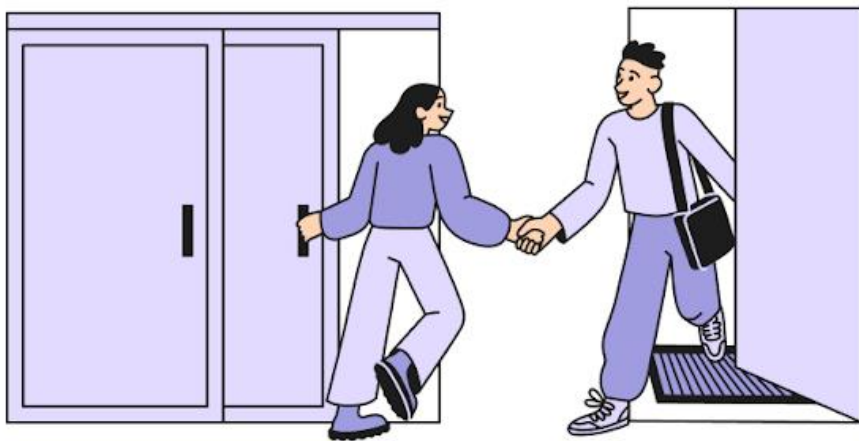
- як влаштована ІТ-галузь
- які бувають проєкти
- яка роль тестувальників і на що вони спираються в роботі.
- познайомимося з основним стандартом галузі – ISTQB сертифікацією.





**Види ІТ компаній**

## Аутсорс



**Аутсорсингові компанії (outsource companies) – компанії, що працюють із сторонніми замовниками. Команди розробників співпрацюють саме з аутсорсинговою компанією, а не з замовником напряму**

## Аутстаф



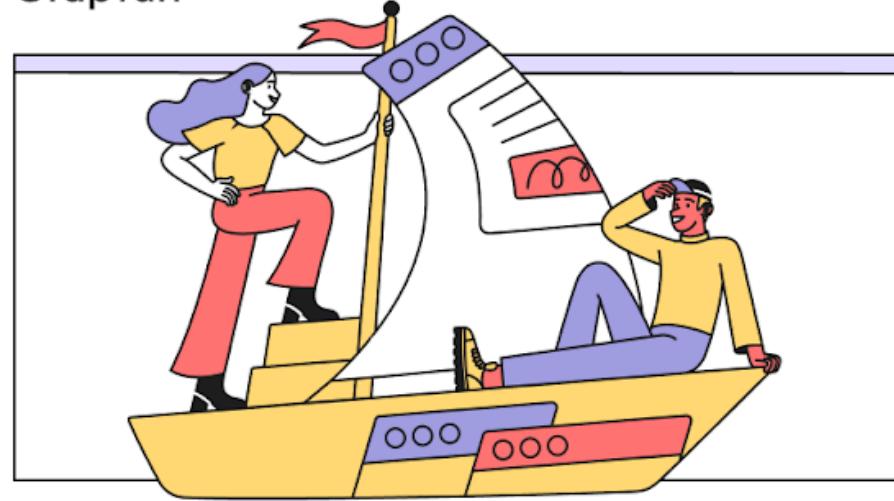
**Аутсорстафінгові компанії (outstaff companies) – компанії, що працюють із сторонніми замовниками. Команди розробників співпрацюють з клієнтом, а компанія лише забезпечує збір команди та організаційні питання**

## Продуктова



**Продуктові компанії – зазвичай такі компанії займаються розробкою власного продукту у вигляді додатку, девайса, сервісу тощо. Продукт (програмне забезпечення) буде власністю компанії і визначатиме технологічний стек та напрямки розвитку.**

## Стартап



**Стартап (Startup) – нещодавно створена компанія, що будує свій бізнес на основі інновацій або інноваційних технологій, не вийшла на ринок або почала на нього виходити і володіє обмеженими ресурсами. Часто стартап-компанії називають «гаражними»**

## Академія



**Компанія, у якій розробники починають свою кар'єру і навчаються, щоб у подальшому перейти до інших компаній**

## Рекрутингове агентство



**Компанія наймає для інших компаній розробників, тобто надає послуги з рекрутингу**



# Програмне забезпечення (software)

- це загальний термін для різних видів програм, що використовуються для роботи з комп'ютерами та пов'язаними з ними пристроями. Програми дають можливість взаємодіяти людині з комп'ютером, забезпечують його зручність і функціональність.

# Види програмного забезпечення

## Системне ПЗ

Призначено для обслуговування власних потреб комп'ютера:

- базовий рівень (firmware) — драйвери
- операційні системи (ОС) — набір програм, які забезпечують взаємодію інших програм з базовими програмами та апаратними засобами
- службовий рівень — програми в складі ОС.

## Прикладне ПЗ

Забезпечує виконання конкретних завдань на комп'ютері: захисту від вірусів, наукових, розважальних та інших. Наприклад текстові та графічні редактори, диспетчери файлів, WEB редактори, архіватори даних, WEB браузері.

## Інструментальне/сервісне ПЗ

ПЗ, призначене для використання в ході створення архітектури, розробки, оновлення та інсталяції програм. Прикладом є середовища розробки.



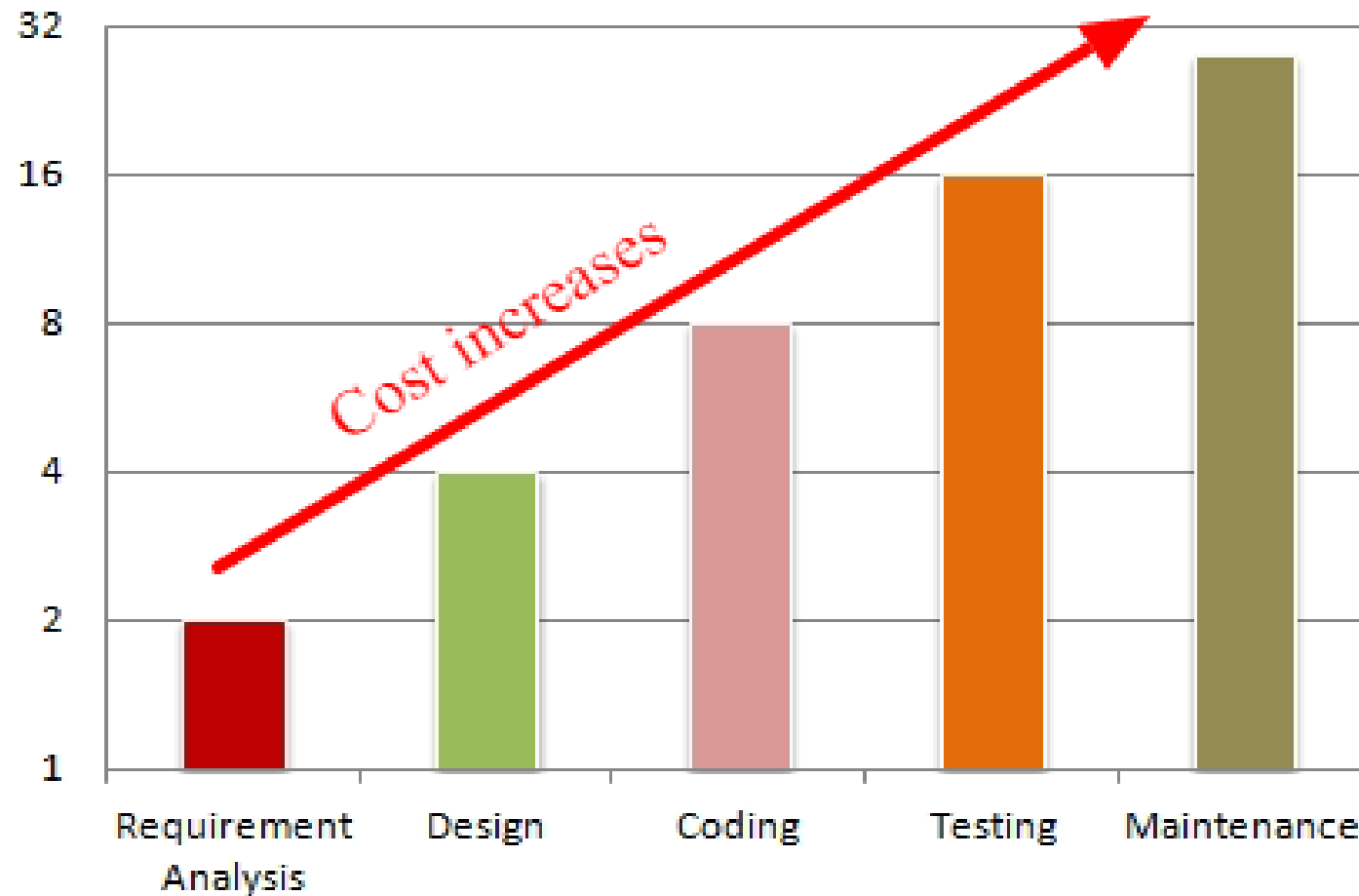




# Якість програмного продукту (software quality)

- це сукупність характеристик, що відносяться до здатності продукту задовольняти встановлені і передбачувані потреби користувача.

# Ціна знайденої помилки



**У 2008 році в новому терміналі британського аеропорту Хітроу встановили суперсучасну систему контролю багажу, призначену для перевезення безлічі вантажів. Система була ретельно протестована з більш ніж 12 тисячами сумок і валіз. У підсумку, в день відкриття терміналу 42000 валіз загубилися, бо не полетіли разом зі своїми господарями. Було скасовано понад 500 авіарейсів.**

**А все тому, що ця нова система не впоралася з реальними сценаріями, які чомусь не були перевірені під час тестування. Наприклад, зняття багажу зі стрічки вручну (пасажир раптово згадав, що там цінні і потрібні йому просто зараз речі) ламало програму, і вона вимикалася.**

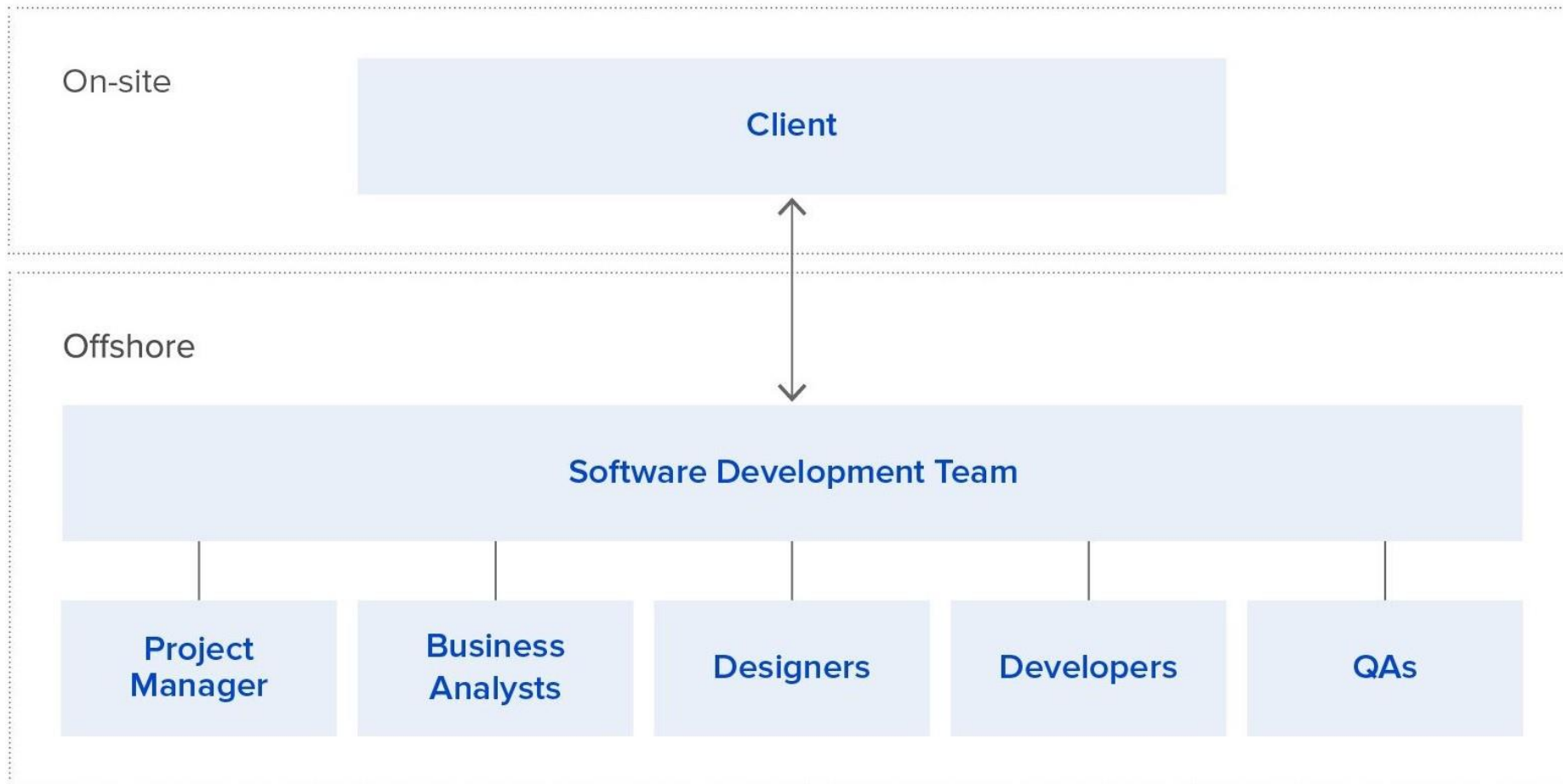
**Відтак, людський фактор найбільше впливає на якість та надійність програм. Можна мільйон разів перевіряти ПЗ на помилки, але завжди є імовірність того, що виникне такий баг, який ламає систему. Як кажуть програмісти-тестери – всього перевірити неможливо, а значить, цілком вірогідно, що саме ви виявити якийсь новий баг.**





**Основні ролі в розробці ПЗ**

# Основні ролі в розробці програмного забезпечення



# QA vs QC vs Testing



# QA vs QC vs Testing

## Забезпечення якості (Quality Assurance - QA)

- це превентивний процес, завданням якого є забезпечення якості продукту в майбутньому. У цьому сенсі Quality Assurance більш орієнтований на процес.

## Контроль якості (Quality Control - QC)

- це процес знаходження помилок в продукті, з метою їх подальшого виправлення. Завданням Quality Control є підтримка якості продукту в поточний момент часу. Quality Control орієнтований на продукт, що розробляється в даний момент.

## Тестування програмного забезпечення (Testing)

- процес перевірки відповідності заявлених до продукту вимог і реально реалізованої функціональності, здійснюваний шляхом спостереження за його роботою в штучно створених ситуаціях і на обмеженому наборі тестів, обраних певним чином







**Навіщо потрібне тестування?**

**Які цілі у тестування?**



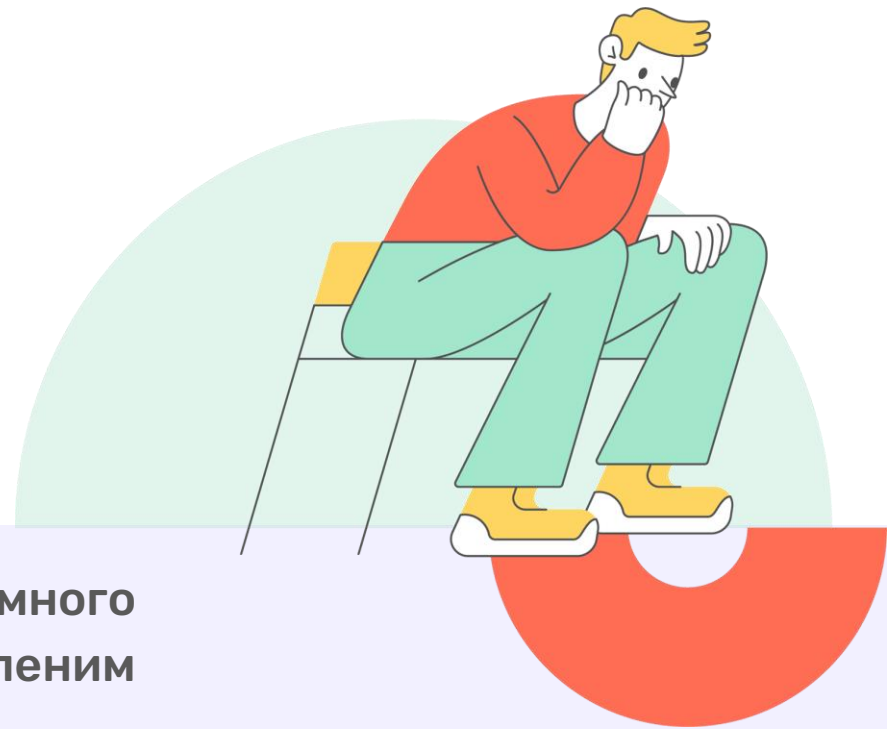
**Помилитись може кожен!**



# Тестування

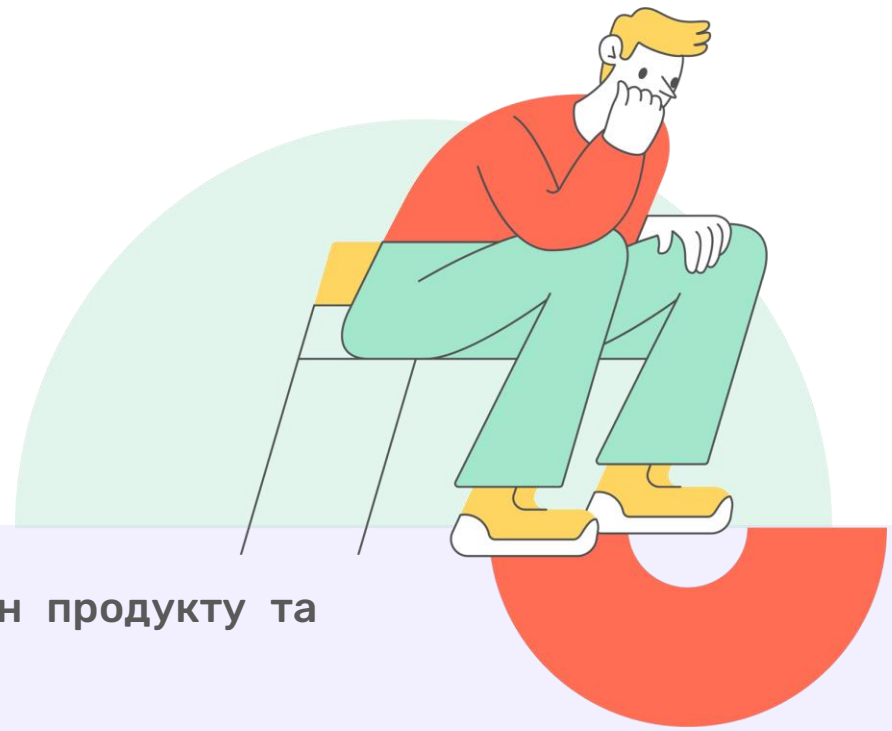
- це аналіз стану і працездатності програмного продукту з метою надання інформації зацікавленим особам (stakeholders)

Визначає відповідність програмного забезпечення вимогам замовника та кінцевого користувача (stakeholders)



# Цілі тестування

- Проаналізувати та надати інформацію про стан продукту та його відповідність вимогам
- Зменшити ризики фінансових, ресурсних втрат та часу
- Локалізувати та виправити критичні помилки
- Спрогнозувати та попередити виникнення помилки
- Віддати користувачу якісний продукт, користування яким приносить задоволення



Тестуємо стілець



# Верифікація – будуємо систему правильно.

- підтвердження на основі подання об'єктивних свідоцтв того, що встановлені вимоги було виконано

# Валідація – будуємо правильну систему.

- підтвердження, що кінцевий продукт відповідає очікуванням і вимогам клієнта. Це динамічний механізм перевірки та тестування фактичного продукту.

# Верифікація та валідація

## VERIFICATION

- 2 sleeves?
- Is it size L?
- Is it blue?
- Are any buttons missing?



## VALIDATION

- Does it fit?
- Is it comfortable to drive in?
- Does the colour match my eyes?
- Can I afford it?
- Is it good quality?
- Will my date like it?

# Верифікація та валідація

I can Verify  
this is a  
house

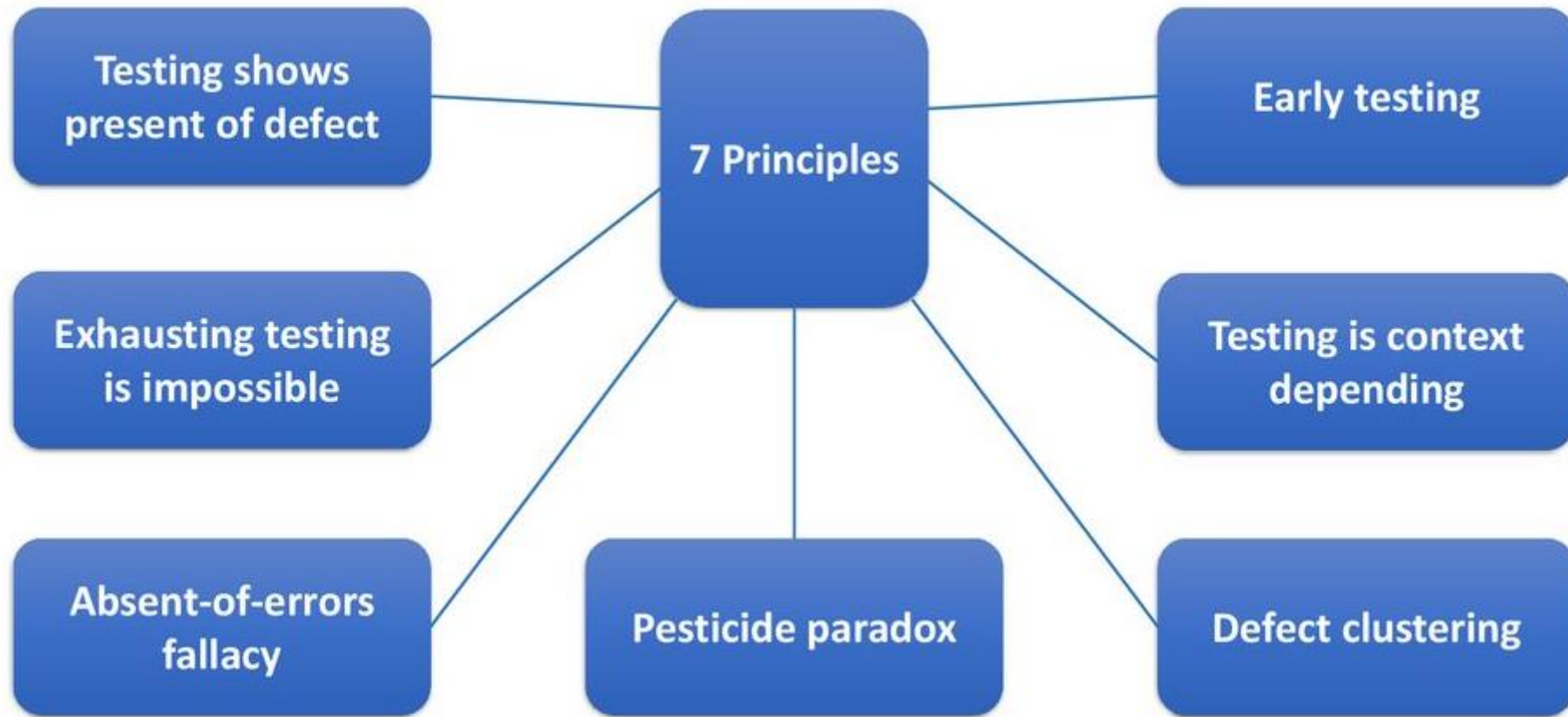


I can Validate that  
this house was  
installed properly





# Принципи тестування





## 1. Тестування виявляє присутність дефектів.

Тестування може показати наявність дефектів у програмі, але не довести їхню відсутність.

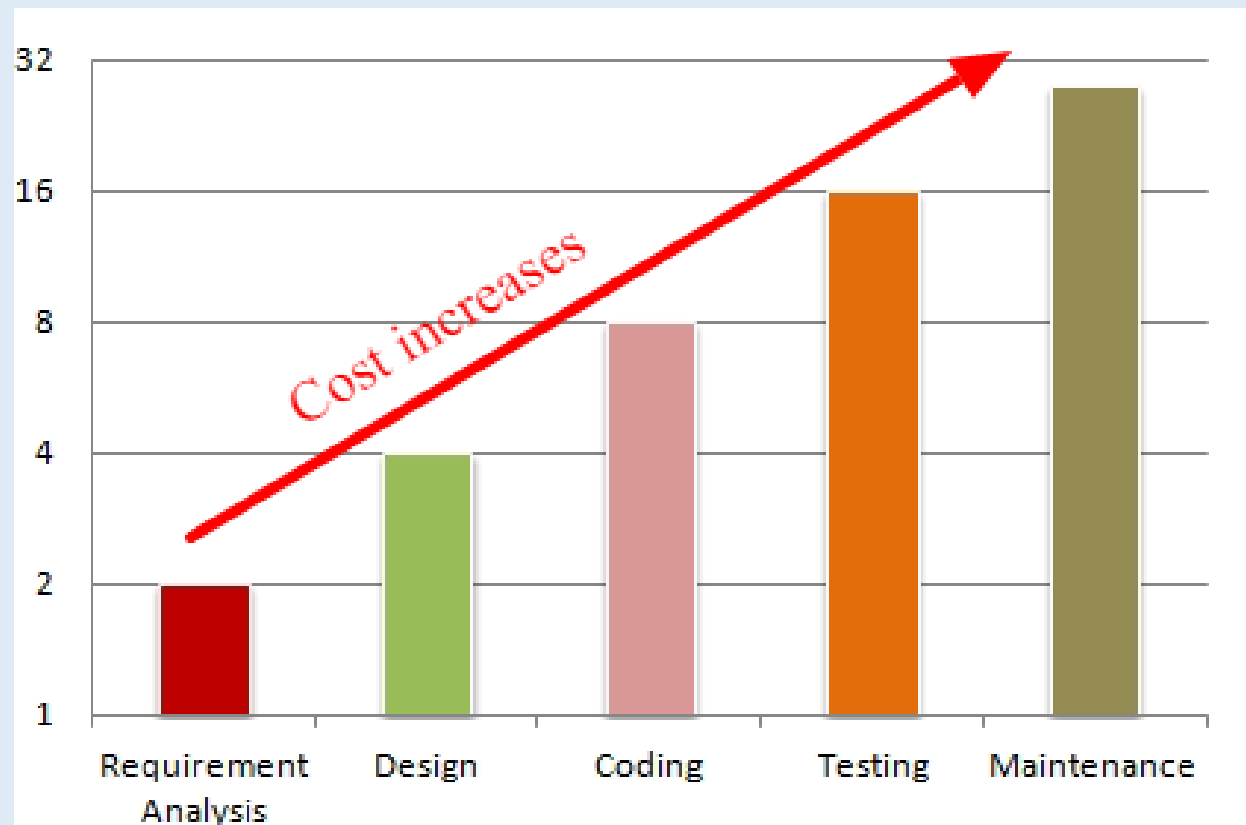
## 2. Вичерпне тестування неможливе.

Неможливо провести вичерпне тестування, яке б покривало всі комбінації введення користувача і станів системи, за винятком зовсім примітивних випадків.



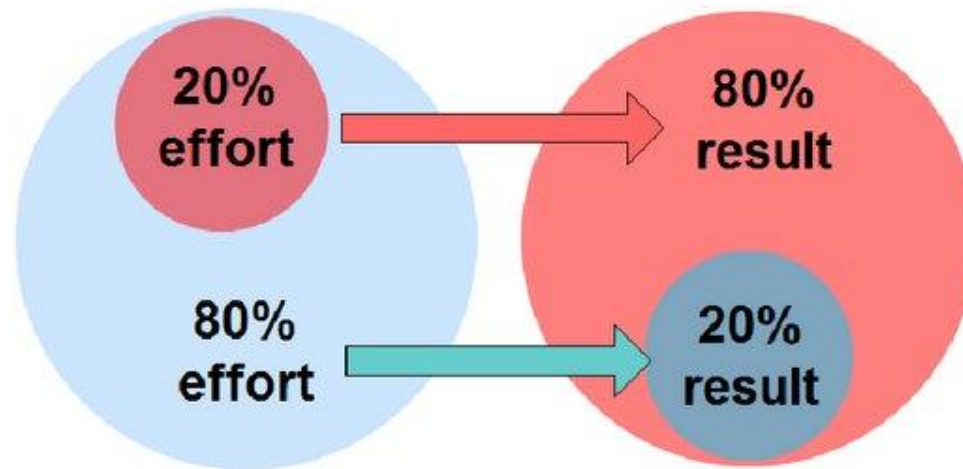
### 3. Раннє тестування.

Тестування має починатися якомога раніше у життєвому циклі розробки програмного забезпечення, та його зусилля мають бути сконцентровані на певних цілях.



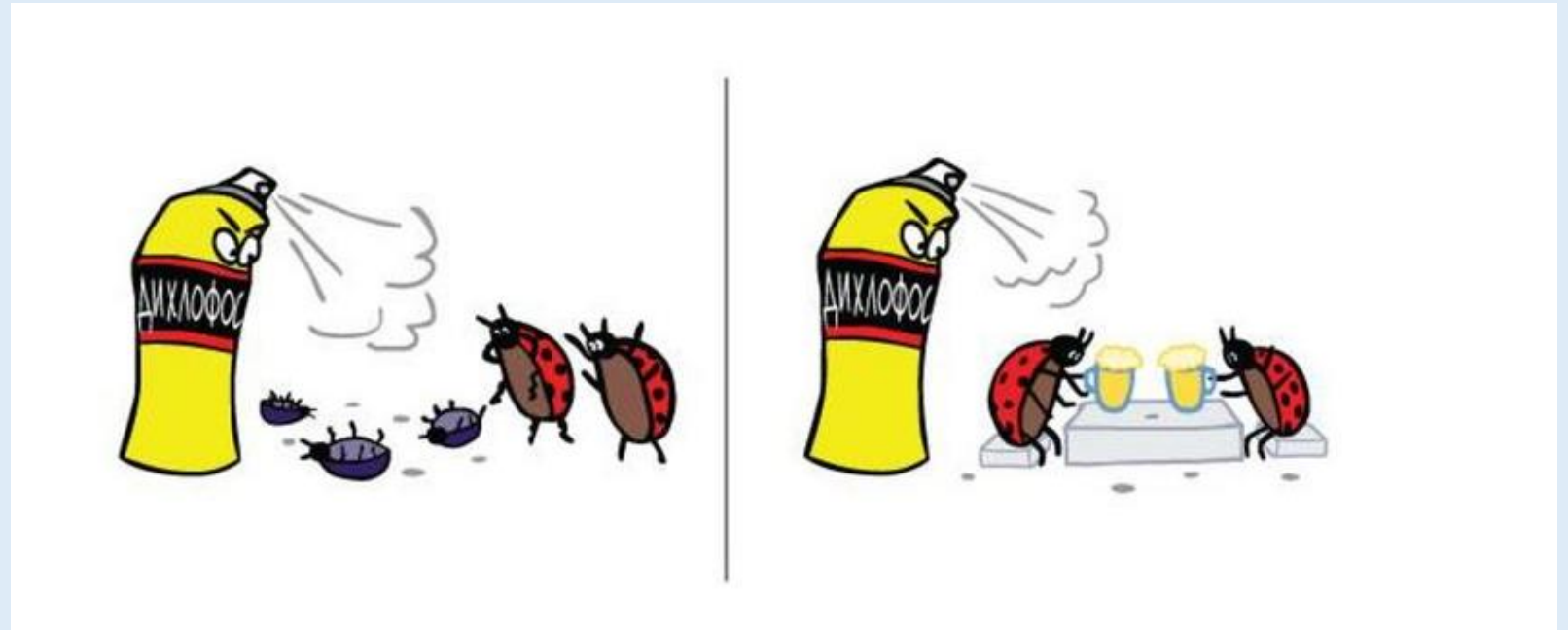
## 4. Групування дефектів.

Різні модулі системи можуть містити різну кількість дефектів - тобто щільність накопичення дефектів в різних елементах програми може відрізнятися. В основному, більшість критичних дефектів знаходять в обмеженій кількості модулів. Це прояв принципу Парето: 80% проблем міститься у 20% модулів.



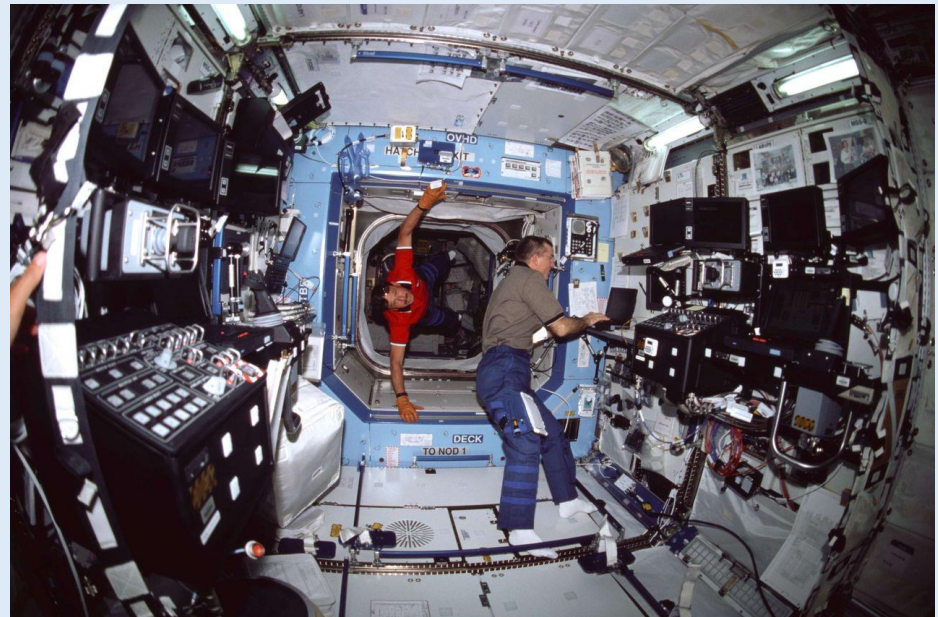
## 5. Парадокс пестицидів.

Проганяючи ті самі тести знову і знову, Ви зіткнетеся з тим, що вони знаходять все менше нових помилок.



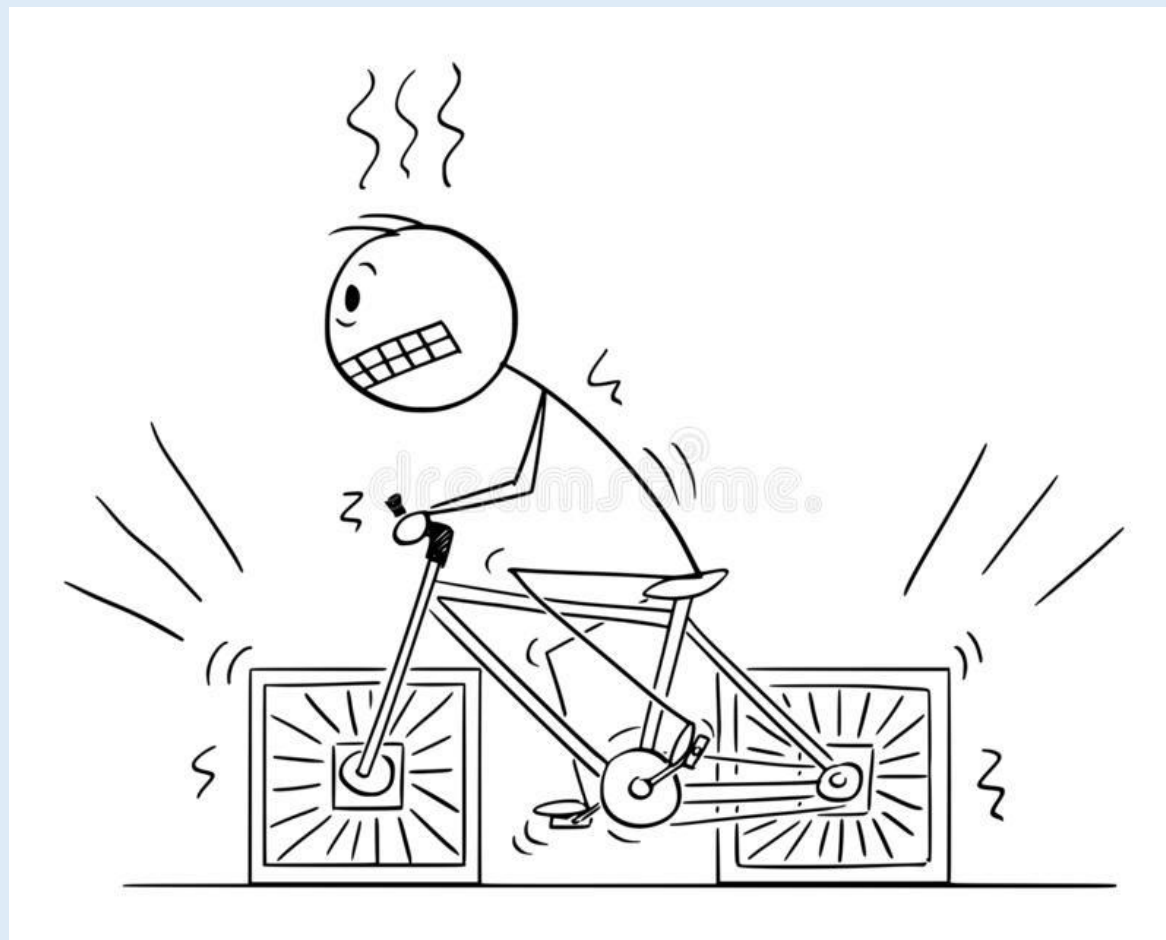
## 6. Тестування залежить від контексту.

Вибір методології, техніки та типу тестування  
безпосередньо залежатиме від природи самої програми.



## 7. Відсутність дефектів оманлива.

Той факт, що тестування не виявило дефектів, ще не означає, що програма готова до релізу. Знаходження та виправлення дефектів будуть не важливими, якщо система виявиться незручною у використанні, і не задовольнятиме очікування та потреби користувача.





ISTQB – некомерційна організація, яка займається питаннями розвитку сфери тестування ПЗ, заснована представниками 8 країн: Австрії, Данії, Фінляндії, Німеччини, Швеції, Швейцарії, Нідерландів та Великобританії.

ISTQB Сертифікація Тестувальника – програма, яка дозволяє спеціалістам отримувати міжнародний сертифікат із тестування.

ISTQB матеріали для скачування - <https://www.rstqb.org/ru/istqb-downloads.html>

ISTQB online glossary - <https://glossary.istqb.org/ru/search/>



