

# Тема 2. Проектування архітектури інформаційних систем

# Що таке архітектура інформаційних систем?

- Система - Сукупність взаємодіючих компонентів та зв'язків між ними, має властивості, яких немає у окремих елементів.
- Інформаційна система (ІС) - сукупність
  - функціональних та інформаційних процесів конкретної предметної області;
  - засобів і методів збору, зберігання, аналізу, обробки і передачі інформації, що залежать від специфіки галузі застосування;
  - методів управління процесами вирішення функціональних завдань, а також інформаційними, матеріальними та грошовими потоками в предметній області

# Системний підхід

- Методологія дослідження об'єктів як систем
- Розкриття цілісності об'єкта
- Виявлення різних типів зв'язків
- Зведення зв'язків у єдину картину

# Забезпечення ІС

- Інформаційне
- Математичне
- Лінгвістичне
- Програмне - сукупність комп'ютерних програм, процедур, алгоритмів, даних та документації; частина ІС, реалізуюча функції обробки інформації, управління даними тощо.
- Технічне - обчислювальна техніка; периферія; засоби зберігання і передачі даних; мережі; засоби введення/виведення
- Організаційне - регламенти, схеми, посадові інструкції тощо
- Нормативно-правове - закони, постанови, накази що регламентують ІС.
- Забезпечення безпеки інформації та даних - конфіденційність, цілісність, доступність, підзвітність і правдивість ресурсів.

# Види інформаційних систем

- За рівнем управління
  - Операційні (транзакційні) системи – автоматизують щоденні рутинні операції (наприклад, системи обліку продажів, банкомати, системи бронювання квитків).
  - Системи управління знаннями – забезпечують збір, збереження та поширення корпоративних знань (бази знань, документообіг).
  - Системи підтримки прийняття рішень (DSS) – допомагають аналізувати дані й моделювати різні варіанти для ухвалення рішень.
  - Системи стратегічного рівня (Executive Information Systems, EIS) – орієнтовані на топ-менеджмент, дають узагальнену інформацію для стратегічних рішень.

# Види інформаційних систем

- За функціональним призначенням
  - Управлінські інформаційні системи (MIS) – формують звітність і аналітику для менеджменту.
  - Інформаційно-довідкові системи – надають доступ до структурованих даних (наприклад, електронні каталоги, довідники).
  - Аналітичні системи (BI – Business Intelligence) – забезпечують глибокий аналіз даних, візуалізації, прогнозування.
  - Експертні системи – базуються на штучному інтелекті, імітують діяльність експерта у вузькій сфері.

# Види інформаційних систем

- За сферою використання
  - Банківські інформаційні системи (наприклад, системи електронних платежів).
  - Освітні інформаційні системи (електронні журнали, Moodle, дистанційне навчання).
  - Медичні інформаційні системи (електронні картки пацієнтів, системи діагностики).
  - Логістичні системи (управління складом, відстеження вантажів).
  - Електронна комерція (інтернет-магазини, платіжні системи).

# Види інформаційних систем

- За способом організації
  - Централізовані – дані зберігаються в одному центрі (наприклад, сервер у компанії).
  - Розподілені – дані та обчислення розподілені між кількома вузлами чи серверами (наприклад, хмарні сервіси).

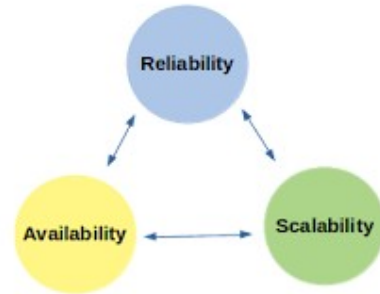
# Види інформаційних систем

- За способом організації
  - Централізовані – дані зберігаються в одному центрі (наприклад, сервер у компанії).
  - Розподілені – дані та обчислення розподілені між кількома вузлами чи серверами (наприклад, хмарні сервіси).

# Вимоги до архітектури ІС

## Функціональні вимоги

- Система повинна виконувати усі бізнес-функції, що закладені замовником.
- Забезпечення інтеграції з іншими системами та сервісами.



## Нефункціональні вимоги

- Надійність – безперервність роботи, стійкість до збоїв, механізми резервування.
- Доступність - частку часу, коли система реально доступна користувачам.
- Продуктивність – оптимальний час відгуку, ефективне використання ресурсів.
- Масштабованість – підтримка зростання кількості користувачів і даних.
- Захищеність – автентифікація, авторизація, шифрування, аудит дій користувачів.
- Ергономічність – зручність інтерфейсу, відповідність вимогам UX/UI.
- Супроводжуваність – простота внесення змін, оновлень та усунення помилок.
- Переносимість – можливість запуску на різних апаратних та програмних платформах.

# Метрики вимірювання якості

- Availability (Доступність)
  - частка часу, коли система доступна користувачам.
  - $\text{Availability} = \text{Uptime} / \text{Total Time} \times 100\%$
  - Приклади SLA (Service Level Agreement): 99.9% ( $\approx 8.7$  год простою на рік), 99.99% ( $\approx 52$  хв/рік).
  - Забезпечується через: кластеризацію, балансування навантаження, резервування.
- 2. Reliability (Надійність)
  - ймовірність, що система виконує функції без збоїв у заданий час.
  - Метрики:
    - MTTF (Mean Time To Failure) – середній час до відмови.
    - MTBF (Mean Time Between Failures) – середній час між відмовами.
    - Failure Rate – частота збоїв.
  - Фокус: не просто щоб система "була доступна", а щоб вона працювала правильно і стабільно.

# Метрики вимірювання якості

- 3. Performance (Продуктивність)
  - Response Time (час відгуку) – середній час відповіді на запит.
  - Throughput (пропускна здатність) – кількість транзакцій/запитів за секунду.
  - Latency (затримка) – затримка між відправкою запиту і початком відповіді.
- 4. Scalability (Масштабованість)
  - Vertical scalability – здатність працювати краще при додаванні ресурсів в один вузол (CPU, RAM).
  - Horizontal scalability – здатність працювати краще при додаванні нових вузлів у кластер.
  - Метрика: коефіцієнт масштабування (ідеально = 1, коли продуктивність росте лінійно з ресурсами).
- 5. Maintainability (Супроводжуваність)
  - наскільки легко підтримувати, оновлювати, виправляти систему.
  - MTTR (Mean Time To Repair) – середній час на відновлення після відмови.
  - Час і вартість внесення змін у код/архітектуру.

# Метрики вимірювання якості

- 6. Security (Безпека)
  - Час реакції на інциденти.
  - Кількість виявлених/усунених вразливостей.
  - Відсоток невдалих спроб входу.
  - Відповідність стандартам (ISO 27001, OWASP Top-10).
- 7. Usability (Зручність використання)
  - Суб'єктивна, але може вимірюватися через:
  - Час, потрібний користувачу для виконання завдання.
  - Кількість кліків/кроків до цілі.
  - Рівень задоволеності (через опитування, NPS).

# Метрики вимірювання якості

- 6. Security (Безпека)
  - Час реакції на інциденти.
  - Кількість виявлених/усунених вразливостей.
  - Відсоток невдалих спроб входу.
  - Відповідність стандартам (ISO 27001, OWASP Top-10).
- 7. Usability (Зручність використання)
  - Суб'єктивна, але може вимірюватися через:
  - Час, потрібний користувачу для виконання завдання.
  - Кількість кліків/кроків до цілі.
  - Рівень задоволеності (через опитування, NPS).

# Історичні періоди розвитку ІС

1950–1960-ті: Початок (обчислювальні центри)

- Використання великих ЕОМ (mainframes) для обчислень і обробки даних.
- Орієнтація на обчислення числових задач (наука, військова сфера).
- Дані зберігались у файлах, кожна програма – ізольована.

Зміни:

- Перші автоматизовані облікові системи.
- З'являється поняття "data processing".

# Історичні періоди розвитку ІС

1970-ті: Базис даних і централізовані ІС

- Виникають реляційні бази даних (Codd, 1970).
- Інформаційні системи стають корпоративними, охоплюють фінанси, управління запасами.
- Зміни:
  - З'являється стандартизація даних.
  - Відділення даних від програм.

# Історичні періоди розвитку ІС

- 1980-ті: Персональні комп'ютери та мережі
  - ПК у компаніях і вдома.
  - Локальні мережі (LAN) з'єднують відділи.

Зміни:

- З'являються перші системи підтримки прийняття рішень (DSS).
- Поняття клієнт-серверної архітектури.

# Історичні періоди розвитку ІС

1990-ті: Інтернет і інтеграція

- Інтернет стає масовим.
- Виникають ERP-системи (SAP, Oracle).

Зміни:

- Потреба інтеграції між різними системами підприємства.
- Стандартизація протоколів обміну (HTTP, XML).

# Історичні періоди розвитку ІС

2000-ні: Веб-сервіси та глобальні системи

- Web 2.0, розквіт електронної комерції.
- З'являються SaaS-рішення (Salesforce, Google Apps).

Зміни:

- Перехід до SOA (Service-Oriented Architecture).
- Глобальна взаємодія через веб-сервіси.

# Історичні періоди розвитку ІС

2010-ті – дотепер: Хмари та мікросервіси

- Cloud computing (AWS, Azure, GCP).
- Мобільні додатки, IoT, AI.

Зміни:

- Розподілені системи на мікросервісах.
- DevOps, CI/CD, контейнеризація (Docker, Kubernetes).
- Великий акцент на scalability, availability, security.

# Властивості ПЗ за Ф.Бруксом

У 1975 р. Фредерік Брукс, проаналізувавши свій досвід керівництва проектом розробки операційної системи OS/360, визначив перелік основних властивостей ПЗ: «складність, узгодженість, змінність і невидимість».

1. Складність. Складність є причиною труднощів, які виникають в процесі спілкування між розробниками, і спричиняє виникнення помилок в продукті, перевищення вартості розробки, затягування процесу виконання графіків робіт. Складність структури ускладнює розвиток ПЗ і додавання нових функцій.

2. Узгодженість. Зазвичай нове ПЗ повинно узгоджуватися із вже існуючим (наприклад, із різними інтерфейсами).

3. Змінність. Програмне забезпечення постійно піддається змінам. При цьому мають місце такі два процеси:

- як тільки виявляють користь програмного продукту, розпочинаються спроби застосування його на межі або за межами первісної області (вимога щодо розширення функцій йде, зазвичай, від користувачів, які задоволені основним призначенням і знаходять для нього нове застосування);
- вдалий програмний продукт живе довше за звичайний термін існування комп'ютера, для якого він спочатку був створений; розробляють нові комп'ютери, і програма повинна бути узгоджена з їх можливостями.

4. Програмний продукт є невидимим. Для матеріальних об'єктів геометричні абстракції є потужним інструментом. План будівлі допомагає архітектору і замовнику оцінити проект у просторі (стають очевидними протиріччя, можна помітити упущення). При спробі графічно зобразити структуру програми виявляється, що потрібно не один, а декілька неорієнтованих графів, накладених один на інший.

Дякую за увагу!