

# Лекція 1: Вступ до веб-орієнтованих систем

Архітектура, проєктування та безпека веб-систем

# Предмет дисципліни

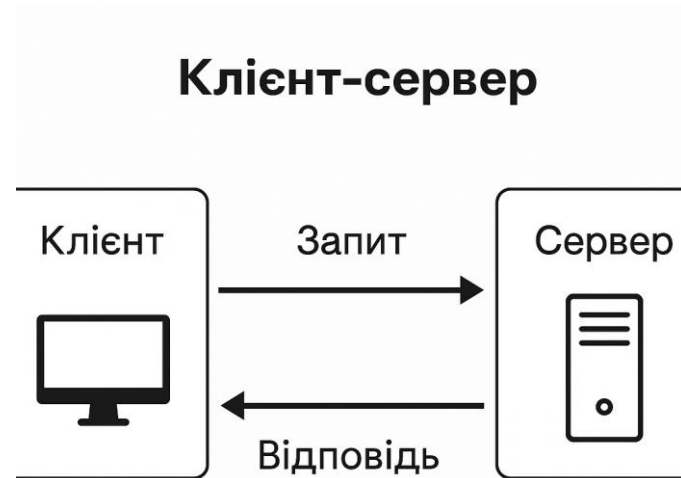
- - Архітектура веб-систем
- - Методи проєктування
- - Принципи безпеки
- - Практичні приклади

# Історія розвитку веб-систем

- - Статичні HTML-сторінки (1990-ті)
- - Динамічні сайти (2000-ні)
- - Веб-додатки (2010-ті)
- - Хмарні та мобільні системи (сьогодні)

# Архітектура клієнт–сервер

- - Клієнт: браузер, мобільний застосунок
- - Сервер: обробка запитів
- - База даних

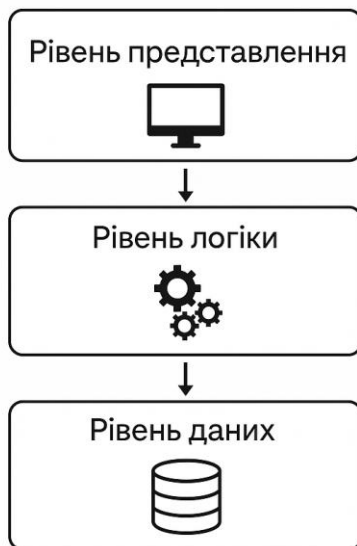


(Схема: клієнт ↔ сервер ↔ база)

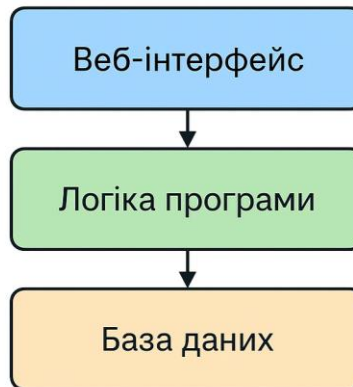
# Багаторівнева архітектура

- - Презентаційний рівень
- - Логічний рівень
- - Рівень даних

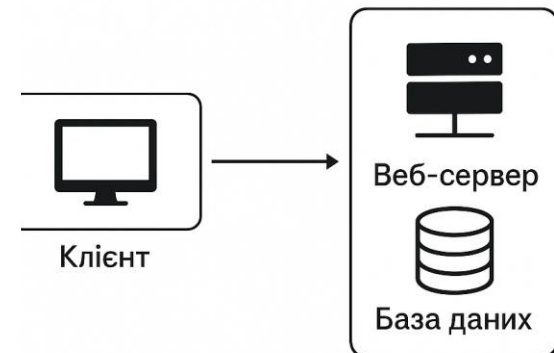
## Багаторівнева архітектура



## Багаторівнева архітектура



## Фізична архітектура веб-системи

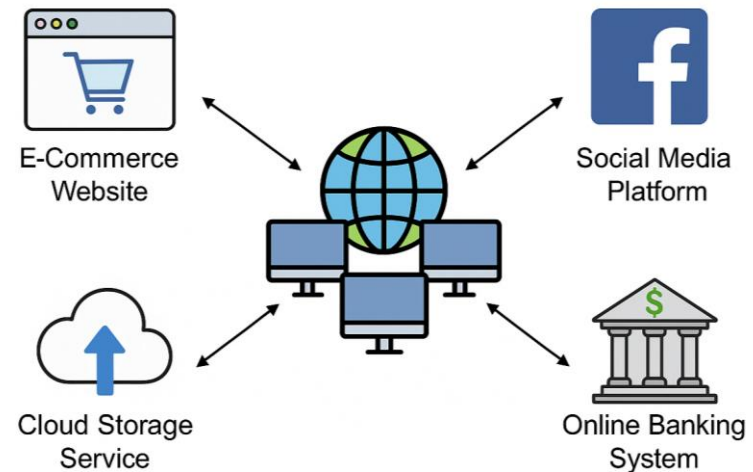


(Приклад: інтернет-магазин)

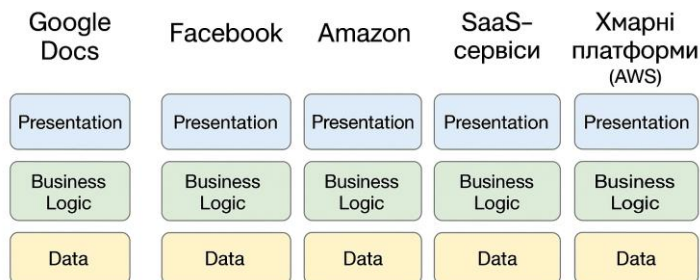
# Сучасні приклади систем

- - Google Docs
- - Facebook
- - Amazon
- - SaaS-сервіси
- - Хмарні платформи

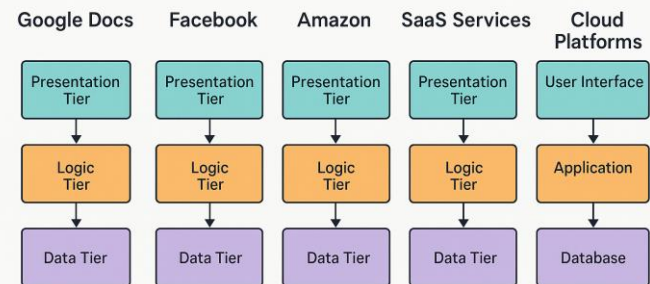
## Сучасні приклади систем



## Сучасні приклади систем



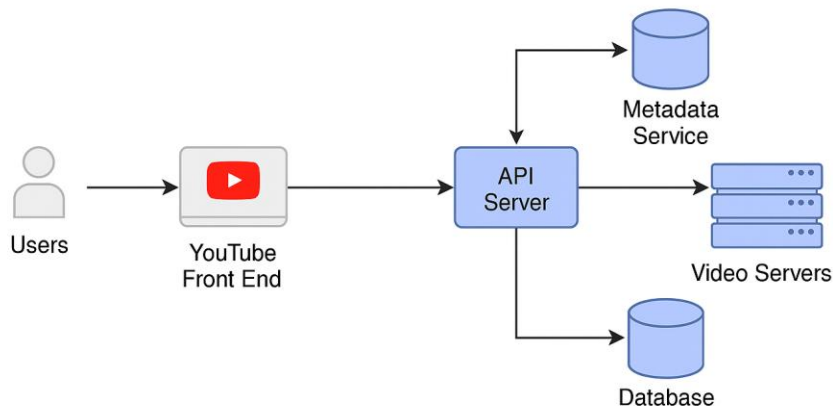
## Сучасні приклади систем



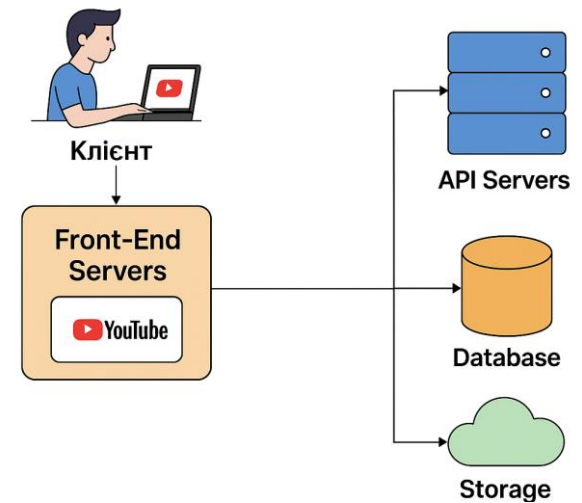
# Кейс: YouTube

- - Клієнт: мобільний застосунок, веб-браузер
- - Сервер: потокове відео, рекомендації
- - База: мільйони відео

Кейс: YouTube



Кейс: YouTube



# Виклики безпеки

- Хакерські атаки.
- Витоки даних.
- Фішинг і соціальна інженерія.
- Внутрішні загрози.



# Роль контейнеризації та віртуалізації

- **Віртуалізація** означає, що ми можемо на одному фізичному сервері запустити кілька віртуальних машин. Кожна з них працює незалежно, як окремий сервер. Це дає економію ресурсів і гнучкість.
- **Контейнеризація** — більш сучасний підхід. Тут ми не створюємо окремі віртуальні машини, а запускаємо контейнери, які містять лише те, що потрібно для роботи застосунку. Вони легші, швидші, простіші в керуванні.

# Питання для самоперевірки

- 1. У чому різниця між клієнтом і сервером?
- 2. Що таке багаторівнева архітектура?
- 3. Які сучасні приклади веб-систем ви знаєте?

# Висновки

- Веб-системи еволюціонували від простих статичних сторінок до складних хмарних платформ.
- Основою є архітектура клієнт–сервер, яка розширюється до багаторівневих і мікросервісних підходів.
- Безпека — це критично важливий аспект, який не можна ігнорувати.
- Сучасні технології, такі як контейнеризація та Docker, змінюють підхід до розробки й розгортання систем.