

ТЕМА 1

Клієнт-серверна архітектура вебу та основи PHP

*HTTP протокол · серверні мови програмування · еволюція вебу · SSR vs CSR
· місце PHP · перший Hello World · XAMPP*

Бекенд-розробка, сервіси та мікросервіси

Комп'ютерна інженерія · 3 курс · Лекція 1 з 18

9 питань сьогоднішньої теми

01

Клієнт-серверна архітектура вебу

02

HTTP протокол детально

03

Серверні мови програмування

04

Еволюція вебу: 1.0 → 2.0 → Mobile

05

Server-Side vs Client-Side Rendering

06

Місце PHP серед бекенд-технологій

07

Коротка історія PHP

08

Синтаксис PHP. Hello World

09

Розгортання XAMPP

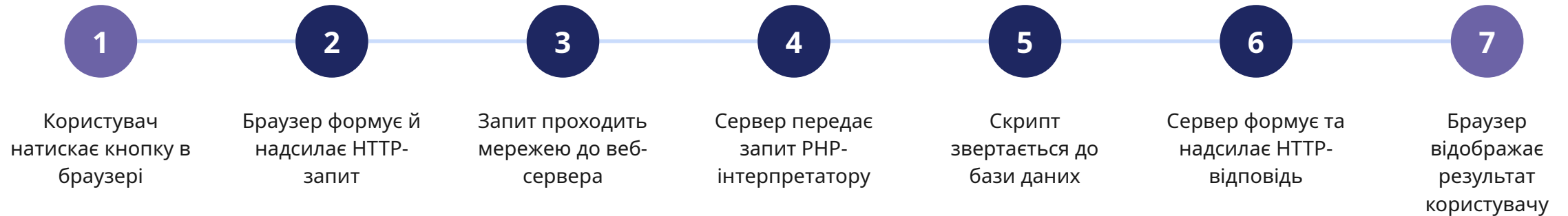
Клієнт запитує — сервер відповідає



Ключова ідея розподілу: клієнт відповідає за представлення, сервер — за дані, бізнес-логіку та їх безпеку. Серверну частину називають **бекендом**, клієнтську — **фронтendom**.

У нашому курсі: рівень представлення (HTML, Vanilla JS) студенти вже знають — тепер будемо рівень бізнес-логіки на PHP.

Життєвий цикл HTTP-запиту



Кожен запит обробляється незалежно від попередніх — HTTP є stateless-протоколом.

Трирівнева (3-tier) архітектура



Що таке HTTP

HTTP (HyperText Transfer Protocol) — прикладний протокол передачі даних між клієнтом і сервером у Вебі. Працює поверх TCP (HTTP/1.1, HTTP/2) або QUIC/UDP (HTTP/3).

HTTP / HTTPS	Формат повідомлень клієнт ↔ сервер
TLS/SSL (лише HTTPS)	Шифрування, цілісність даних
TCP / QUIC	Надійна доставка пакетів
IP	Маршрутизація пакетів мережею

Stateless: сервер не пам'ятає користувача між запитами — для цього використовують cookies і сесії.

Структура HTTP-запиту

```
GET /catalog/products?category=laptops HTTP/1.1
Host: shop.example.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64)
Accept: text/html,application/xhtml+xml
Accept-Language: uk-UA,uk;q=0.9
Cookie: session_id=a1b2c3d4
Connection: keep-alive
```

■ Рядок запиту

Метод, шлях (URI), версія протоколу

■ Заголовки

Метадані: клієнт, мова, куки, тип контенту

■ Тіло (body)

Дані форми — відсутнє в GET-запитах

HTTP-методи

Метод	Призначення	Тіло
GET	Отримати ресурс	Немає
POST	Створити ресурс / надіслати форму	Є
PUT	Повністю оновити ресурс	Є
PATCH	Частково оновити ресурс	Є
DELETE	Видалити ресурс	Зазвичай немає
HEAD	Як GET, але без тіла відповіді	Немає
OPTIONS	Дізнатись підтримувані методи (CORS)	Немає

GET, HEAD, OPTIONS, PUT, DELETE — ідempотентні: повторний виклик не змінює результат.

Коди статусів HTTP

1xx	Інформаційні	100 Continue
2xx	Успіх	200 OK · 201 Created · 204 No Content
3xx	Перенаправлення	301 · 302 · 304 Not Modified
4xx	Помилка клієнта	400 · 401 · 403 · 404 Not Found
5xx	Помилка сервера	500 · 502 Bad Gateway · 503

Ключові заголовки та еволюція версій

Ключові заголовки

Content-Type MIME-тип тіла (text/html, application/json)

Authorization Облікові дані (Basic, Bearer-токен)

Cookie / Set-Cookie Передача та встановлення кук

Cache-Control Правила кешування ресурсу

Еволюція версій протоколу

HTTP/1.1

1997

Постійні з'єднання (keep-alive)

HTTP/2

2015

Мультиплексування, стиснення заголовків

HTTP/3

2022

Поверх QUIC/UDP, менші затримки

Огляд поширених бекенд-технологій

PHP

Веб-сайти, CMS (WordPress), API

Низький поріг входу

Node.js (JS)

API реального часу, чати

Той самий синтаксис, що й на фронті

Python

Дата-сервіси, ML-інтеграції

Лаконічний синтаксис

Java

Корпоративні, банкінг

Продуктивність JVM

C# (.NET)

Корпоративні застосунки

Екосистема Microsoft

Go

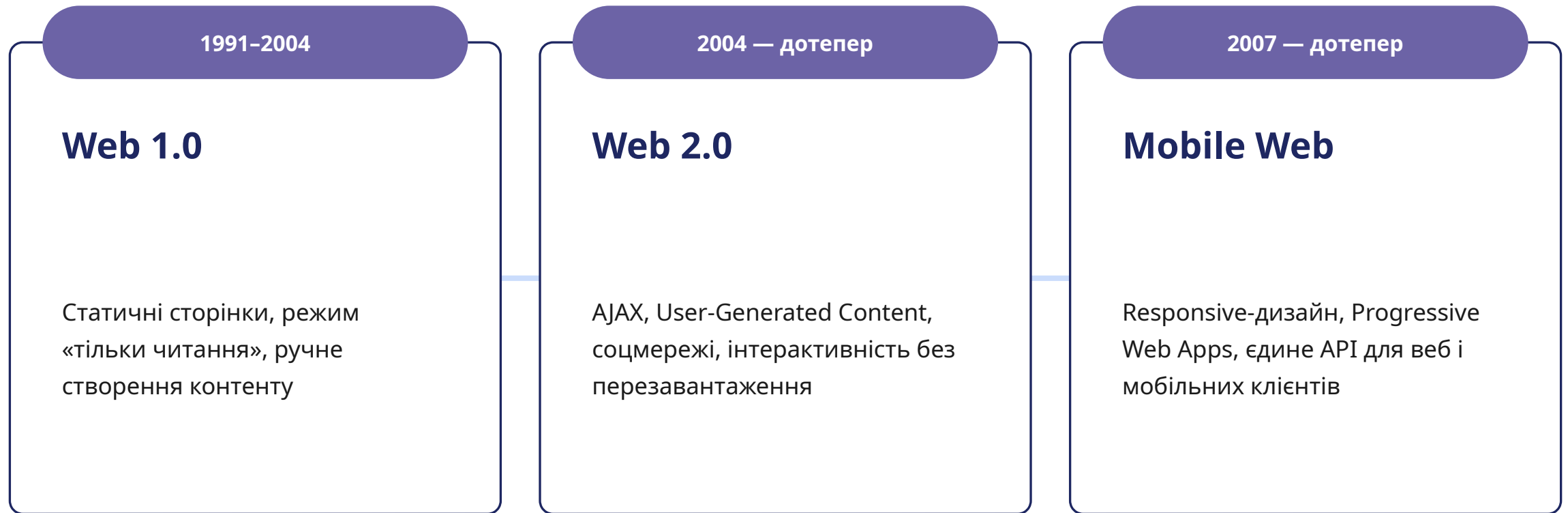
Мікросервіси, високі навантаження

Вбудована паралельність

Чому саме PHP у цьому курсі

- 1 Мова створювалася спеціально для Вебу
- 2 Мінімальний поріг входу — перший скрипт за хвилини
- 3 Понад 70% ринку серверних технологій
- 4 Сучасний PHP 8.x — повноцінна ООП-мова з типізацією
- 5 Логічний місток від HTML/Vanilla JS до бекенду

Web 1.0 → Web 2.0 → Mobile Web



Server-Side Rendering vs Client-Side Rendering

SSR

Server-Side Rendering

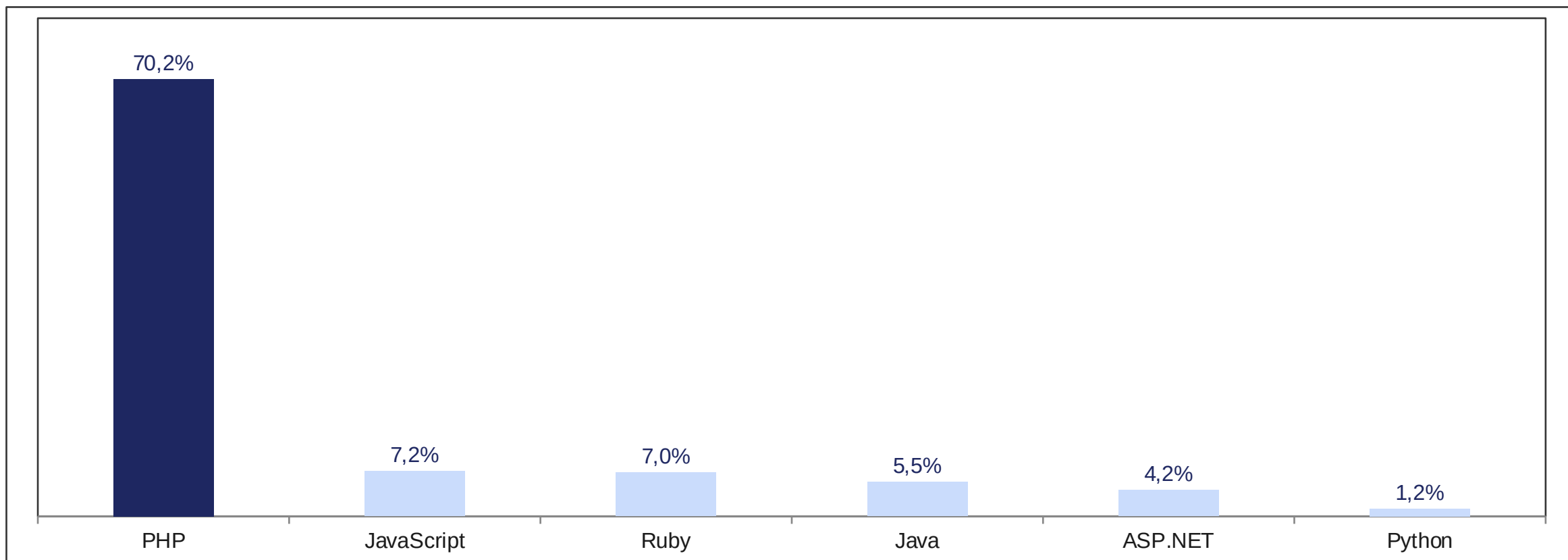
- Сервер формує готовий HTML одразу
- Швидше перше відображення сторінки
- Простіша індексація (SEO)
- Вище навантаження на сервер
- Класичний підхід: WordPress, PHP-сайти

CSR

Client-Side Rendering

- Браузер сам будує інтерфейс через JS
- Сервер віддає дані через JSON API
- Природна взаємодія без перезавантажень (SPA)
- Складніша індексація без доп. технік
- Підхід: React/Vue/Vanilla JS + PHP API

PHP використовує 70,2% сайтів з відомою серверною мовою



Джерело: W3Techs, вересень 2026 · PHP 8 нині встановлено на більшості активних PHP-сайтів

Від скрипта для лічильника відвідувань до PHP 8

- 
- A vertical timeline with a light blue line and dark blue circular markers at each point. The years are listed on the left, and the corresponding PHP milestones are listed on the right.
- 1994-95** ● Rasmus Lerdorf створює «Personal Home Page Tools»
 - 1998** ● PHP 3 — мова переписана «з нуля»
 - 2000** ● PHP 4 — рушій Zend Engine
 - 2004** ● PHP 5 — повноцінна підтримка ООП
 - 2015** ● PHP 7 — продуктивність зростає у 2-3 рази
 - 2020** ● PHP 8.0 — JIT-компіляція, атрибути, union-типи
 - 2023-26** ● PHP 8.2-8.4 — типізація, readonly-класи, enum

Перша програма: Hello World

```
<!DOCTYPE html>
<html lang="uk">
<head><meta charset="UTF-8"><title>Перша PHP-сторінка</title></head>
<body>

    <?php
        echo "Hello, World!";
    ?>

</body>
</html>
```

Код між **<?php ... ?>** виконується сервером; браузер отримує лише готовий HTML і ніколи не бачить сам PHP-код.

ХАМРР: усе для локальної розробки в одному дистрибутиві

X

крос-
платформність

Windows / macOS / Linux

A

Apache

Веб-сервер

M

MariaDB / MySQL

Сервер бази даних

P

PHP

Інтерпретатор

P

Perl

*Рідше
використовується в
курсі*

Структура проєкту: тека htdocs

```
htdocs/  
├─ index.php  
├─ hello-world/  
│   └─ index.php  
└─ lab1/  
    └─ task1.php
```

htdocs/index.php

→ <http://localhost/>

hello-world/index.php

→ <http://localhost/hello-world/>

lab1/task1.php

→ <http://localhost/lab1/task1.php>

Файл потрібно відкривати через <http://localhost/...>, а не подвійним клацанням у Провіднику — інакше PHP-код виконано не буде.

Підсумок теми 1

- Клієнт-серверна модель розділяє представлення й бізнес-логіку
- HTTP — stateless-протокол запитів і відповідей
- PHP займає понад 70% ринку серверних мов
- SSR формує HTML на сервері, CSR — у браузері
- XAMPP розгортає Apache + MySQL + PHP за кілька хвилин