



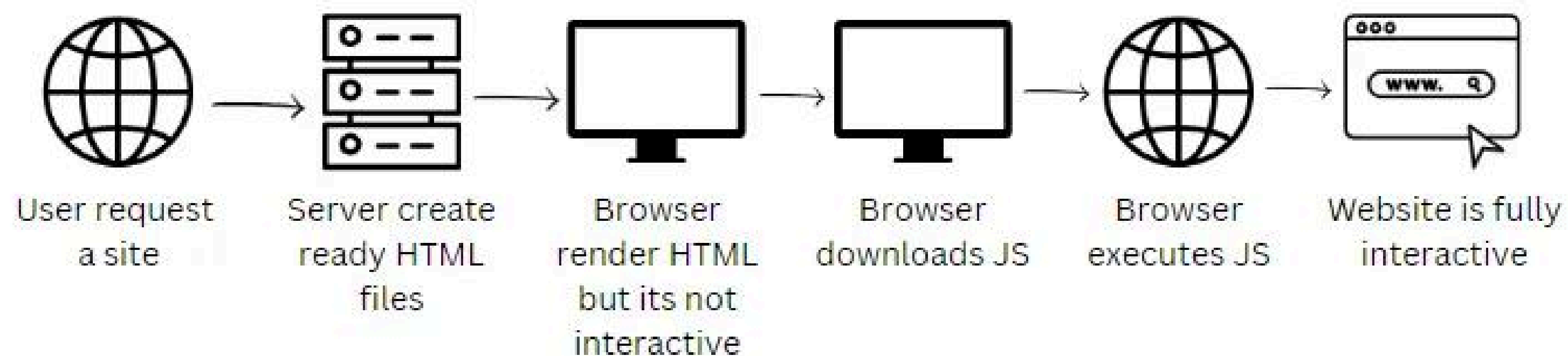
Серверный рендеринг. Шаблонизация

“ИТ-класс”. Бэкенд-разработка на Python
Бусыгин Дмитрий, 2025 год

Что такое SSR?

SSR (Server-Side Rendering) — страница формируется на сервере и отправляется браузеру уже готовой

SSR (Server-side Rendering)



Client-side vs Server-side

Клиентский рендеринг

1. Браузер получает пустой HTML
2. Загружает JavaScript
3. JS создаёт содержимое

- ✓ Интерактивность
- ✓ Динамический контент
- ✗ Медленная первая загрузка
- ✗ Проблемы с SEO

Серверный рендеринг

1. Сервер создаёт HTML
2. Отправляет готовую страницу
3. Браузер сразу показывает

- ✓ Быстрая первая загрузка
- ✓ Отличное SEO
- ✓ Работает без JS
- ✗ Требуется сервер

Реализация на Python

Jinja2 - движок шаблонов для Python

Позволяет создавать HTML-страницы с динамическим содержимым, используя Python-переменные и логику



Возможности Jinja2

- Вставка переменных
- Условные операторы
- Циклы
- Фильтры
- Наследование шаблонов
- Макросы и функции

Jinja2 шаблон

```
<h1>Привет, {{ username }}!</h1>
```

Обычный HTML

```
<h1>Привет, Пользователь!</h1>
```

Базовый пример

Шаг 1: Настройка в FastAPI

```
from fastapi import FastAPI, Request
from fastapi.templating import Jinja2Templates

app = FastAPI()
templates = Jinja2Templates(directory="templates")

@app.get("/")
async def home(request: Request):
    return templates.TemplateResponse(
        "index.html",
        {"request": request, "title": "Главная страница"}
    )
```

Шаг 2: Создание шаблона

```
←!— templates/index.html →
<!DOCTYPE html>
<html>
<head>
    <title>{{ title }}</title>
</head>
<body>
    <h1>Добро пожаловать!</h1>
    <p>Это страница: {{ title }}</p>
</body>
</html>
```

Копаем глубже

Условия (if)

```
←!— Условия →
{% if user.age ≥ 18 %}
<p>Вы совершеннолетний</p>
{% elif user.age ≥ 14 %}
<p>Вы подросток</p>
{% else %}
<p>Вы ребёнок</p>
{% endif %}
←!— Проверка на существование →
{% if user.email %}
<p>Email: {{ user.email }}</p>
{% endif %}
```

Полезные переменные в цикле

loop.index — номер итерации (с 1)

loop.index0 — номер итерации (с 0)

loop.first — первая итерация

loop.last — последняя итерация

Циклы (for)

```
←!— Цикл for →
<h2>Список предметов:</h2>
<ul>
{% for subject in subjects %}
  <li>{{ subject }}</li>
{% endfor %}
</ul>
←!— Цикл с индексом →
<ol>
{% for subject in subjects %}
  <li>Предмет #{{ loop.index }}: {{ subject }}</li>
{% endfor %}
</ol>
←!— Пустой список →
<ul>
{% for item in items %}
  <li>{{ item }}</li>
{% else %}
  <li>Список пуст</li>
{% endfor %}
</ul>
```

Фильтры для переменных

```
←!— Фильтры для строк →
<p>{{ name|upper }}</p> ←!— АЛЕКСАНДР →
<p>{{ name|lower }}</p> ←!— александр →
<p>{{ name|title }}</p> ←!— Александр →
<p>{{ name|capitalize }}</p> ←!— Александр →

←!— Фильтры для списков →
<p>{{ subjects|length }}</p> ←!— 3 →
<p>{{ subjects|first }}</p> ←!— Математика →
<p>{{ subjects|last }}</p> ←!— Физика →
<p>{{ subjects|join(", ") }}</p> ←!— Математика, Информатика, Физика →

←!— Фильтры для чисел →
<p>{{ price|round(2) }}</p> ←!— 99.99 →
<p>{{ 42|abs }}</p> ←!— 42 →
←!— Значение по умолчанию →
<p>{{ email|default("Не указан") }}</p>
←!— Безопасный HTML →
<div>{{ html_content|safe }}</div>
```

```
←!— Цепочка фильтров →
<p>{{ text|lower|title }}</p>
←!— Фильтры в циклах →
{% for subject in subjects|sort %}
<li>{{ subject|upper }}</li>
{% endfor %}
```

Популярные фильтры

upper - в верхний регистр

lower - в нижний регистр

length - длина списка/строки

join - объединить список

default - значение по умолчанию

safe - не экранировать HTML

Наследование шаблонов

```
<!-- templates/base.html -->
<!DOCTYPE html>
<html>
  <head>
    <title>{% block title %}Мой сайт{% endblock %}</title>
  </head>
  <body>
    <header>
      <h1>Школьный портал</h1>
      <nav>
        <a href="/">Главная</a>
        <a href="/about">О нас</a>
      </nav>
    </header>
    <main>
      {% block content %}
    {% endblock %}
    </main>
    <footer>
      <p>© 2025 Школа</p>
    </footer>
  </body>
</html>
```

```
<!-- templates/home.html -->
{% extends "base.html" %}
{% block title %}Главная страница{% endblock %}
{% block content %} <h2>Добро пожаловать!</h2>
<p>Это главная страница нашего сайта.</p>  <ul> {%
for news in news_list %} <li>{{ news.title }}</li>
{% endfor %} </ul> {% endblock %}
```

Итог: одни шапка и подвал на весь сайт.
Не надо их писать для каждой
страницы отдельно

**А теперь посмотрим
на SSR в действии**