

Что такое SSR!

(Server Side Rendering)

Какие есть ещё **Rendering**

Зачем нам это нужно

Как это работает

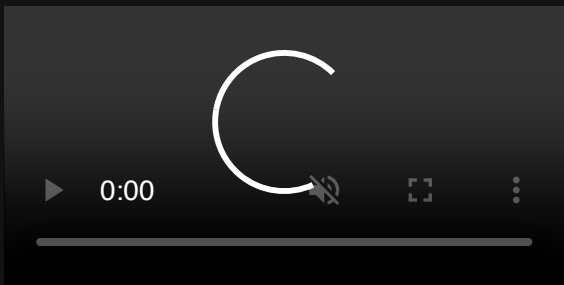
inertia»
SSR

PAKETA

 **Laravel**
Blade Templates

Что бы понять в чём приколот SSR

Нужно понимать какие есть ещё **Rendering** и в чём их различия

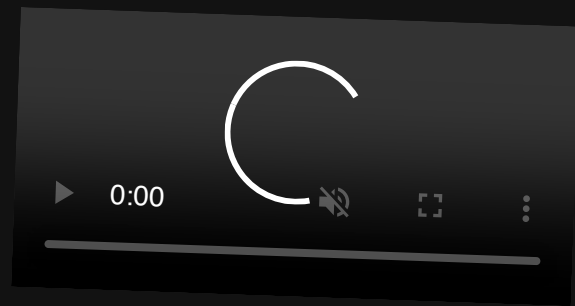
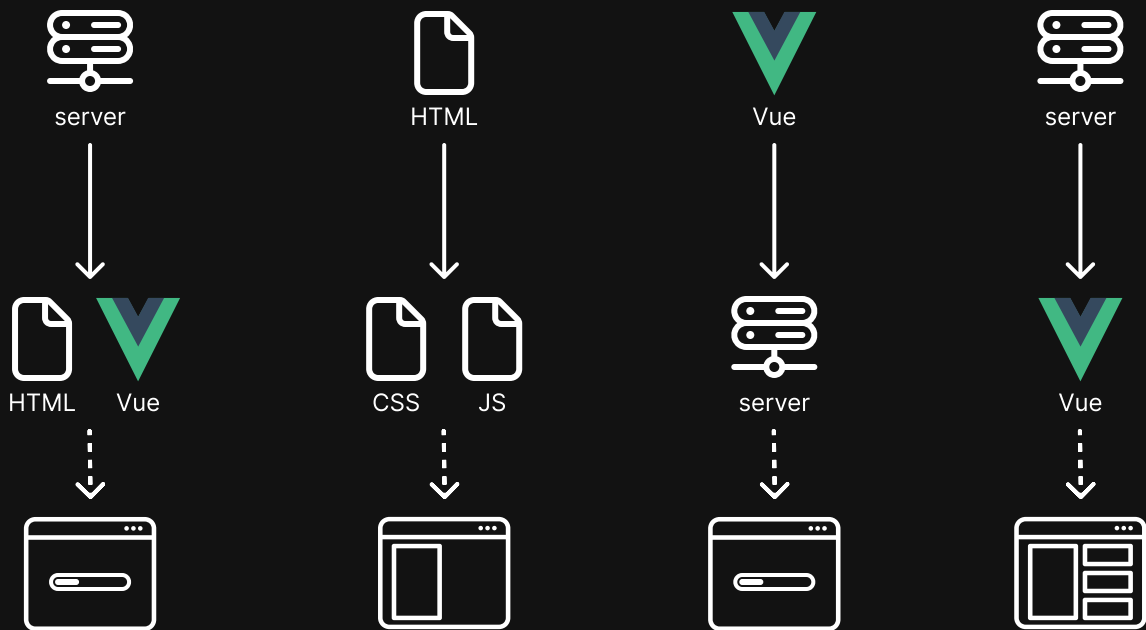


- **SSG** (Static Site Generation)
- **CSR** (Client Side Rendering)
- **SSR** (Server Side Rendering)

Генератор статики **SSG** (Static Site Generation)

Очень редкий зверь, но невероятно прикольная штука. По сути это генератор html страниц который может собрать многостраничный сайт без бека)

Как это работает: (Допустим нужно вывести список победителей с фильтром и поиском)



Плюсы минусы SSG

Генератор статических страниц

■ Плюсы ■

■ Минусы ■

Нету бека

Нету бека

Скорость загрузки

SEO Friendly (если 100% статика)

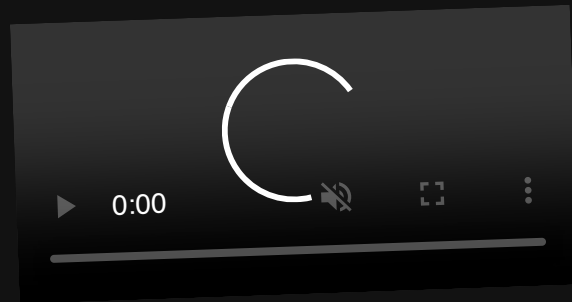
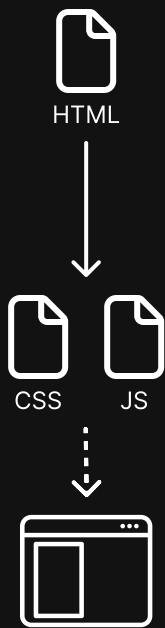
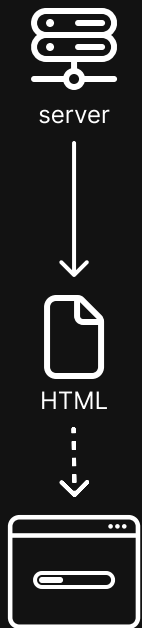
Легко развернуть и настроить

Ссылки: [Astro](#) | [11ty](#) | [Gatsby.js](#)

Старый добрый **CSR** (Client Side Rendering)

Мы все этим пользуемся когда используем js фреймворки или библиотеки

Как это работает: (Тоже выводим список победителей)



Плюсы минусы CSR

Это по сути любые js фреймворки

Плюсы

Легко развернуть и настроить

Минусы

SEO

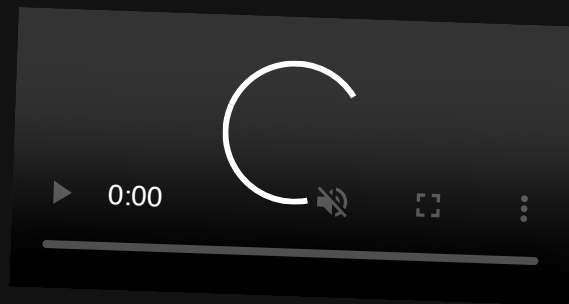
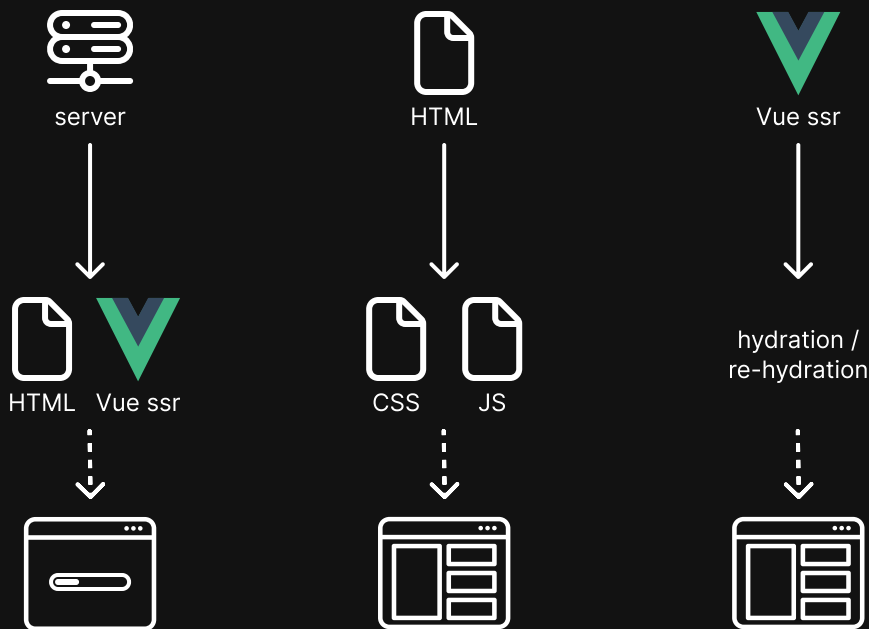
Проблемы с архитектурой

Ссылки: [Svelte](#) | [Vue.js](#) | [React.js](#)

И сигма из всей тусы **SSR** (Server Side Rendering)

Формируем страницу до того как клиент её увидит!

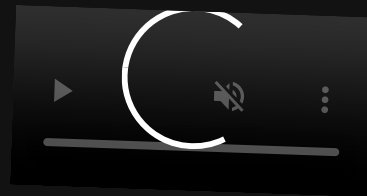
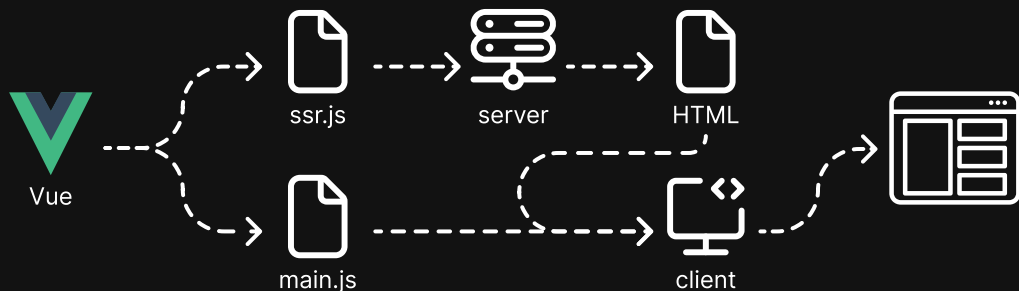
Как это работает: (Тоже выводим список победителей)



Как работает SSR ?

По сути мы просто крутим JavaScript на сервере, который частично или полностью выполняет работу бекенда

html генерируется на сервере с помощью Bun.js или Node.js , JavaScript формируется во время сборки проекта. Из за этого клиентский JavaScript знает где и какой html нужно ожидать и что с ним делать дальше. Это называется hydration / re-hydration



Важно учитывать что часть js выполняется на сервере а часть на клиенте

Плюсы минусы SSR

Laravel + js фреймворки и никаких `.blade.php`

■ Плюсы ■

SEO Friendly

Удобно организовать архитектуру

Полное разделение бек фронт

Бек на Laravel

■ Минусы ■

Сложная настройка

Современные сервера

Ссылки: [Inertia.js](#)

Зачем нам нужен SSR ?

Спойлер: это нужно только для seo ну, и немного для производительности но если ты wb или ozon

- Можно полностью писать фронт на js фреймворках без каких либо ограничений
- На прямую передавать данные в js фреймворк во момент формирования страницы на сервере
- Вся логика формирования html на стороне сервера, но только при первой загрузки страницы

Если во всём этом разобраться то это очень удобно и быстро разрабатывается и масштабируется