



NILS HARTMANN

<https://nilshartmann.net>

Practical
introduction:
Fullstack
with
Next.js

INTERNATIONAL JAVASCRIPT CONFERENCE | MUNICH, NOVEMBER, 13 2024 | @NILSHARTMANN

NILS HARTMANN

nils@nilshartmann.net

Freelance Software Developer, Architect and Trainer from Hamburg

Java, Spring, GraphQL, React, TypeScript



<https://graphql.schule/video-kurs>



<https://reactbuch.de>

[HTTPS://NILSHARTMANN.NET](https://nilshartmann.net)

Beispiel-Anwendung

Source-Code (GitHub):

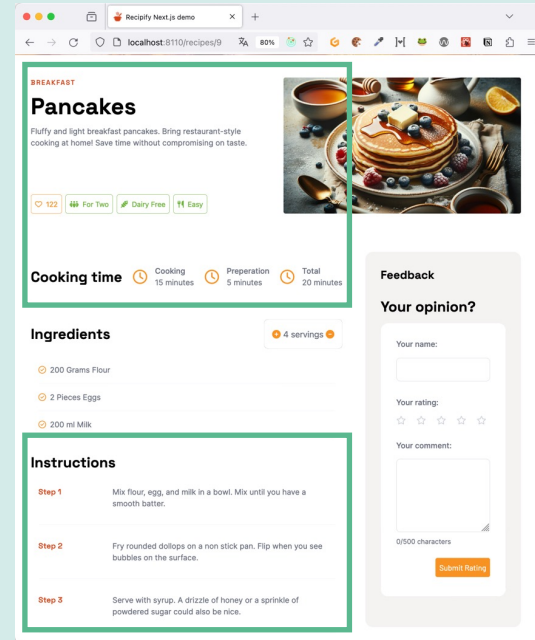
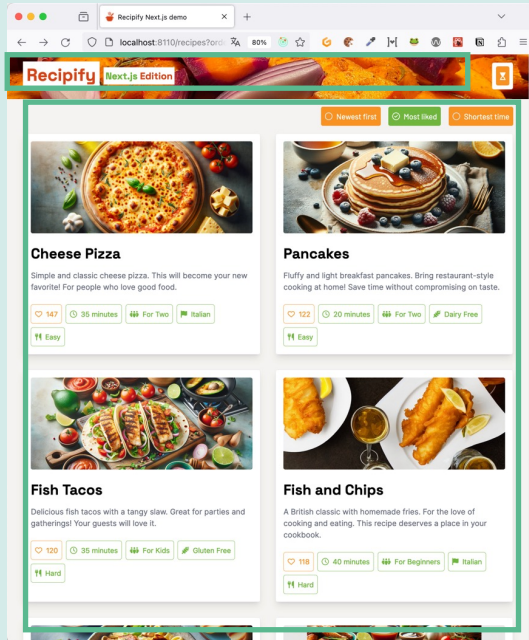
<https://react.schule/nextjs-beispiel>

Nach dem Vortrag checke ihr hier den geschriebenen Code ein
(eigener Branch)

EIN BEISPIEL

Was macht die Beispiel-Anwendung aus?

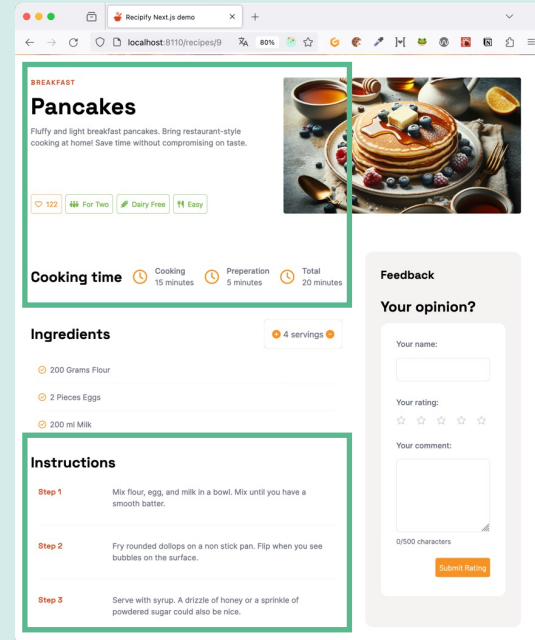
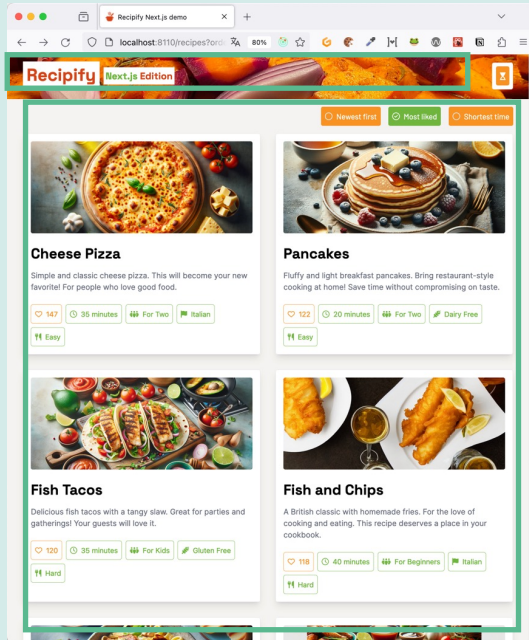
- Viel Essen 🍴



EIN BEISPIEL

Was macht die Beispiel-Anwendung aus?

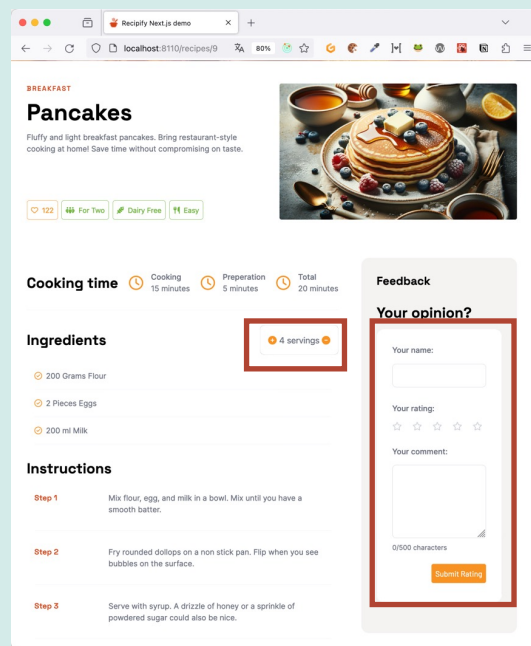
- Viel statischer Content 😊



EIN BEISPIEL

Was macht die Beispiel-Anwendung aus?

- Viel statischer Content 😊
- ... wenig Interaktion 🦥
- ... does that justify tons of JS in the client? 🤔



EIN BEISPIEL

Was macht die Beispiel-Anwendung aus?

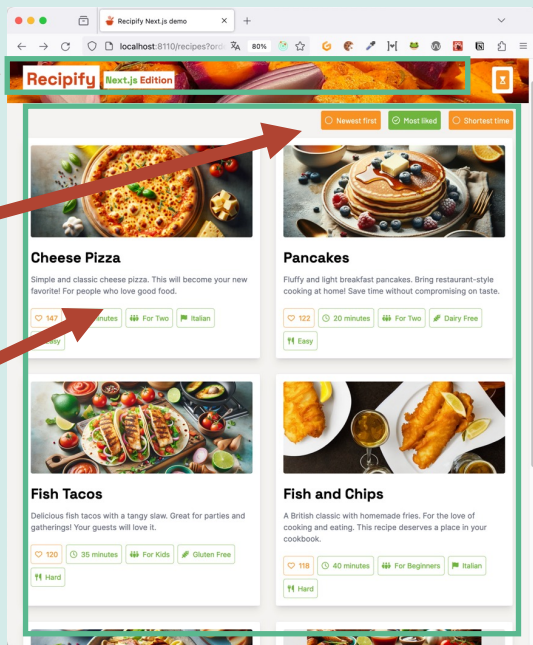
- ...aber viel JavaScript-Code 🤖

React!

Header.tsx

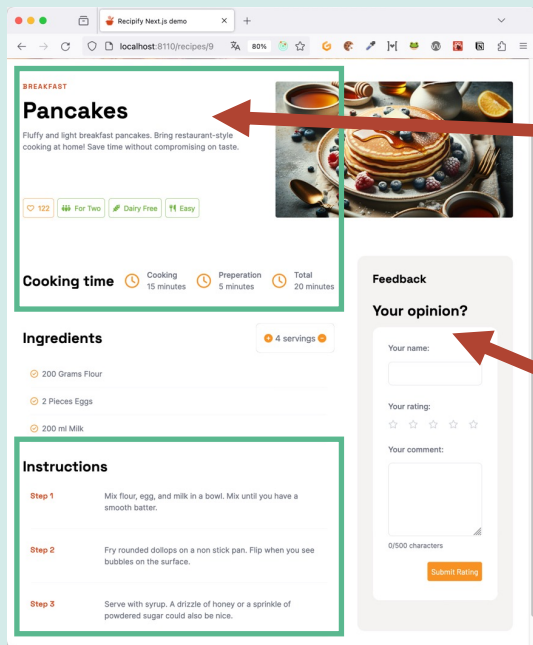
Nav.tsx

RecipeList.tsx



Banner.tsx

Feedback.tsx



Mögliche Probleme

- (Viel) JavaScript-Code, der...

Mögliche Probleme

- (Viel) JavaScript-Code, der...
 - ... vom Browser geladen werden muss

Mögliche Probleme

- (Viel) JavaScript-Code, der...
 - ... vom Browser geladen werden muss
 - ... interpretiert und ausgeführt werden muss

Mögliche Probleme

- (Viel) JavaScript-Code, der...
 - ... vom Browser geladen werden muss
 - ... interpretiert und ausgeführt werden muss
- ...und mit jeder neuen Komponente mehr wird

Go full-stack with a framework

React is a library. It lets you put components together, but it doesn't prescribe how to do routing and data fetching. To build an entire app with React, we recommend a full-stack React framework like [Next.js](#) or [Remix](#).

<https://react.dev/>

"Fullstack Architektur-Vision"

<https://react.dev/learn/start-a-new-react-project#which-features-make-up-the-react-teams-full-stack-architecture-vision>

"Fullstack Architektur-Vision"

<https://react.dev/learn/start-a-new-react-project#which-features-make-up-the-react-teams-full-stack-architecture-vision>

- **React Server Components (RSC):**
 - Komponenten, die auf dem Server, Client und im Build gerendert werden können
 - Data Fetching "integriert"

"Fullstack Architektur-Vision"

<https://react.dev/learn/start-a-new-react-project#which-features-make-up-the-react-teams-full-stack-architecture-vision>

- **React Server Components (RSC):**

- Komponenten, die auf dem Server oder im Build gerendert werden können
- Data Fetching "integriert"

- **Suspense:**

- Platzhalter für "langsame" Teile einer Seite
- Mit Streaming können diese Teile einer Seite "nachgeliefert" werden, sobald sie gerendert sind

React empfiehlt "Fullstack-Framework"

- **Server Components** erfordern Rendern auf dem Server oder im Build
- Dazu braucht man ein "**Fullstack-Framework**"

React empfiehlt "Fullstack-Framework"

- **Server Components** erfordern Rendern auf dem Server oder im Build
- Dazu braucht man ein "**Fullstack-Framework**"
- "**Framework**" ist verharmlosend, weil es sich in der Regel um einen kompletten Stack samt Build-Tools und Laufzeitumgebung handelt

React empfiehlt "Fullstack-Framework"

- **Next.js** entspricht den Vorstellungen des React-Teams

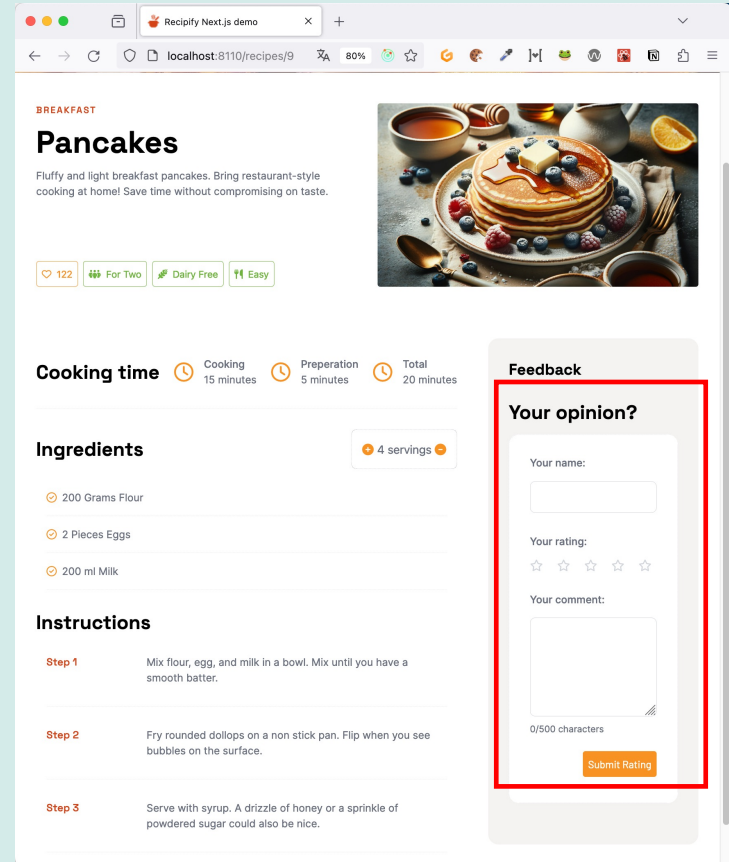
Zero-Bundle-Size
Server
Components

Arten von Komponenten

ARTEN VON KOMPONENTEN

Client-Komponenten (wie bisher)

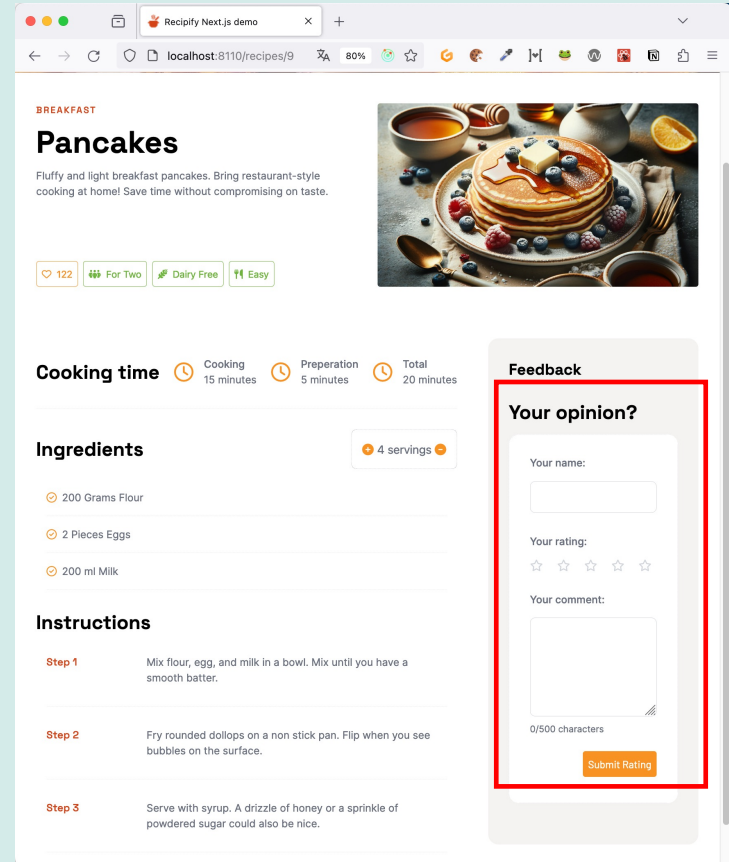
- Werden auf dem Client gerendert



ARTEN VON KOMPONENTEN

Client-Komponenten (wie bisher)

- Werden auf dem Client gerendert
- oder auf dem Server 🤔



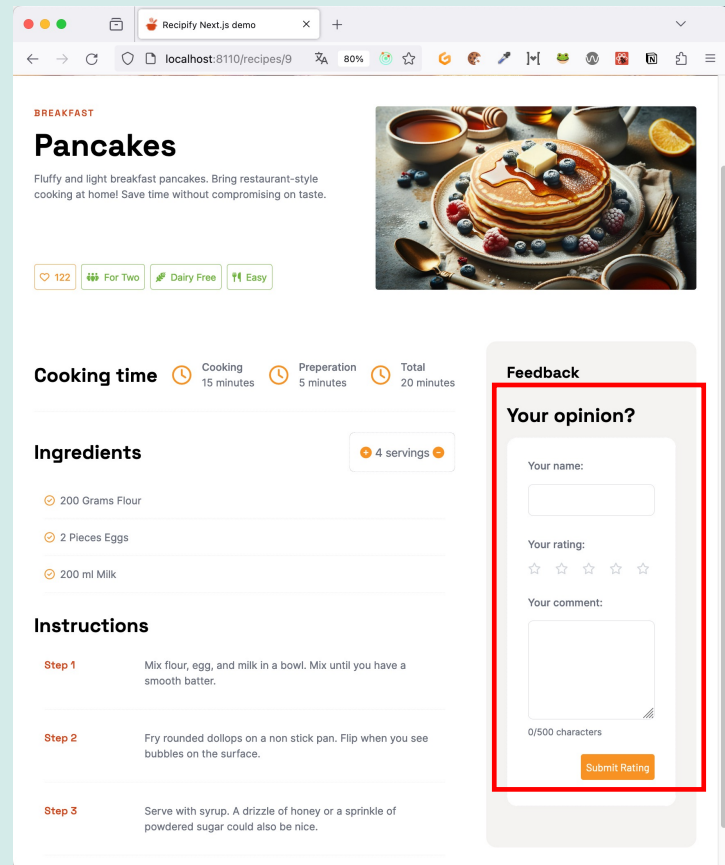
ARTEN VON KOMPONENTEN

Client-Komponenten (wie bisher)

- Werden auf dem Client gerendert
- oder auf dem Server 🤔

Wie bisher:

- JavaScript-Code immer zum Client gesendet
- Können deshalb interaktiv sein



Neu: Server-Komponenten

- werden auf dem Server gerendert

Neu: Server-Komponenten

- werden auf dem Server gerendert
- oder im Build 🙄

Neu: Server-Komponenten

- werden auf dem Server gerendert
- oder im Build 🤖
- liefern UI (!) zum React-Client zurück (kein JavaScript-Code)

ARTEN VON KOMPONENTEN

Komponenten können gemischt werden



RSC am Beispiel Next.js

NILS HARTMANN

<https://nilshartmann.net>

Thank you very much!

Slides & Code: <https://react.schule/ijs-nextjs>

Questions and contact:

nils@nilshartmann.net

<https://nilshartmann.net/contact>

