

NILS HARTMANN

<https://nilshartmann.net>

The background of the slide features a photograph of a coastal scene. In the foreground, a small, colorful boat is moored in shallow, blue water. Behind it, a larger, dark-colored fishing boat is beached on a rocky, sandy shore. The sky is clear and blue.

Moderne Web-Frontends: Single-Page-Anwendung oder HTMX?

NILS HARTMANN

<https://nilshartmann.net>

Moderne
Web-Frontends:
Single P

Planänderung!

HTMX?

NILS HARTMANN

<https://nilshartmann.net>

Alternativen zur Single-Page-Awendung: JS-Fullstack oder HTMX?

NILS HARTMANN

nils@nilshartmann.net

Freiberuflicher Entwickler, Architekt, Trainer

Java, Spring, GraphQL, React, TypeScript



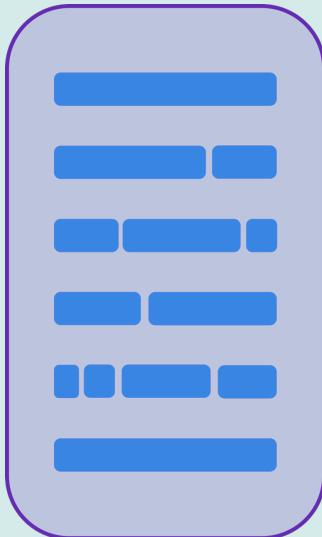
<https://graphql.schule/video-kurs>

<https://reactbuch.de>

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

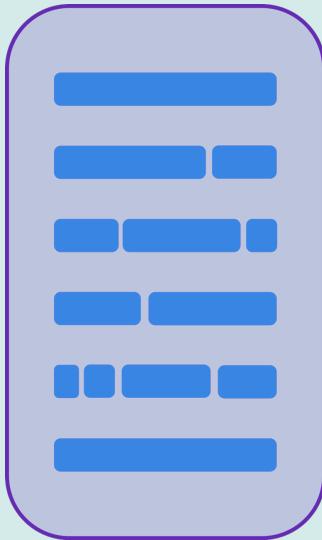
Architekturen für Webanwendungen

Single-Page-Anwendungen



- Darstellung erfolgt vollständig mit JavaScript
- Statisches HTML spielt (fast) keine Rolle
- Die Anwendung kommuniziert mit dem Backend über API
- Ausgetauscht werden Daten, aber keine UI
- Vertreter: Angular, **React**, Svelte, Vue

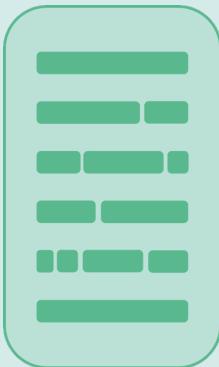
Single-Page-Anwendungen: Eure Meinung!



- Wo machen SPA Sinn? 🤔
- Welche Probleme gibt es damit? 🤯
- Gängige Vorurteile? 😡

Alternativen zur SPA

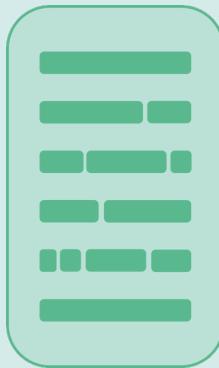
"Klassisch": Serverseitiges Rendern



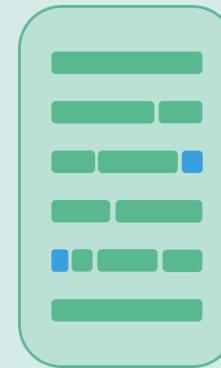
HTML-Seite

- Server liefert fertige HTML-Seite zum Client
- Für jede Interaktion dann Server-Roundtrip (Link, Formular)
- Typische Vertreter: PHP, Java, dotNET, ...

"Klassisch": Serverseitiges Rendern plus JavaScript "Schnipsel"



HTML-Seite



...mit JavaScript-Schnipseln

- Für feingranulare Updates wird JavaScript eingestreut

"Klassisch": Serverseitiges Rendern plus JavaScript "Schnipsel"



- Eigentlich optimal:
 - wir haben **JavaScript** nur da, wo wir es **wirklich** brauchen, für Interaktivität
 - alles andere kann statisches HTML und CSS sein ❤️

"Klassisch": Serverseitiges Rendern plus HTMX



HTML-Seite

- Statt JavaScript Schnipsel nehmen wir **HTMX**

HTMX ist eine JavaScript-Bibliothek

introduction

htmx gives you access to [AJAX](#), [CSS Transitions](#), [WebSockets](#) and [Server Sent Events](#) directly in HTML, using [attributes](#), so you can build [modern user interfaces](#) with the [simplicity](#) and [power](#) of [hypertext](#)

htmx is small (~14k min.gz'd), [dependency-free](#), [extendable](#), IE11 compatible & has [reduced](#) code base sizes by 67% when compared with react

<https://htmx.org/>

Versprechen: Aktualisierung der Darstellung ohne selbst JavaScript **schreiben**
zu müssen
(stattdessen "hypertext" 🤯)

HTMX - Grundlagen

Demo: Hello World, HTMX!

<http://localhost:9080/hello>

```
<html lang="en">
  <head>
    <title>Hello World</title>
    <script src="/htmx/htmx-1.9.10.min.js"></script>
  </head>
  <body>
    <div id="result"></div>
    <button type="button" htmx-target="#result">Get Greeting</button>
  </body>
</html>
```

HTMX - Grundlagen

- HTML-Elemente werden mit HTMX-Attributen ergänzt

```
<html lang="en">
  <body>
    <div hx-get="/hello-world"
          hx-trigger="click"
          hx-target="#result">
      Get Greeting
    </div>

    <div id="result"></div>

    <script
      type="text/javascript"
      th:src="@{/htmx/htmx-1.9.10.min.js}"
    ></script>
  </body>
</html>
```

HTMX - Grundlagen

- HTML-Elemente werden mit HTMX-Attributen ergänzt
- Damit wird beschrieben, welche Server Requests bei einem "Trigger" ausgeführt werden sollen

```
<html lang="en">
  <body>
    <div hx-get="/hello-world"
          hx-trigger="click"
          hx-target="#result">
      Get Greeting
    </div>

    <div id="result"></div>

    <script
      type="text/javascript"
      th:src="@{/htmx/htmx-1.9.10.min.js}"
    ></script>
  </body>
</html>
```

HTMX - Grundlagen

- HTML-Elemente werden mit HTMX-Attributen ergänzt
- Damit wird beschrieben, welche Server Requests bei einem "Trigger" ausgeführt werden sollen
- HTMX kümmert sich um die Ausführung des Requests und die Verarbeitung der Antwort

```
<html lang="en">
  <body>
    <div hx-get="/hello-world"
          hx-trigger="click"
          hx-target="#result">
      Get Greeting
    </div>

    <div id="result"></div>

    <script
      type="text/javascript"
      th:src="@{/htmx/htmx-1.9.10.min.js}"
    ></script>
  </body>
</html>
```

HTMX – Der Server: Anforderungen

- Der Server nimmt (Ajax-)Requests entgegen

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class HelloWorldController {

    @GetMapping(value = "/hello-world",
                headers = "HX-Request")
    String helloWorldResponse() {
        return "hello-world-response";
    }
}
```

HTMX – Der Server: Anforderungen

- Der Server nimmt (Ajax-)Requests entgegen
- HTMX-Requests sind per HTTP Header zu identifizieren

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class HelloWorldController {

    @GetMapping(value = "/hello-world",
                headers = "HX-Request")
    String helloWorldResponse() {
        return "hello-world-response";
    }
}
```

HTMX – Der Server: Anforderungen

- Der Server nimmt (Ajax-)Requests entgegen
- HTMX-Requests sind per HTTP Header zu identifizieren
- Der Server liefert dann HTML-Schnipsel

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class HelloWorldController {

    @GetMapping(value = "/hello-world",
                headers = "HX-Request")
    String helloWorldResponse() {
        return "hello-world-response";
    }
}
```

HTMX – Der Server: Anforderungen

- Der Server nimmt (Ajax-)Requests entgegen
- HTMX-Requests sind per HTTP Header zu identifizieren
- Der Server liefert dann HTML-Schnipsel
- Wie die Schnipsel erzeugt werden, entscheidet der Server bzw. ihr

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class HelloWorldController {

    @GetMapping(value = "/hello-world",
                headers = "HX-Request")
    String helloWorldResponse() {
        return "hello-world-response";
    }
}
```

HTMX – Der Server: Anforderungen

- Der Server nimmt (Ajax-)Requests entgegen
- HTMX-Requests sind per HTTP Header zu identifizieren
- Der Server liefert dann HTML-Schnipsel
- Wie die Schnipsel erzeugt werden, entscheidet der Server bzw. ihr
- Zum Beispiel mit **Template-Sprache**

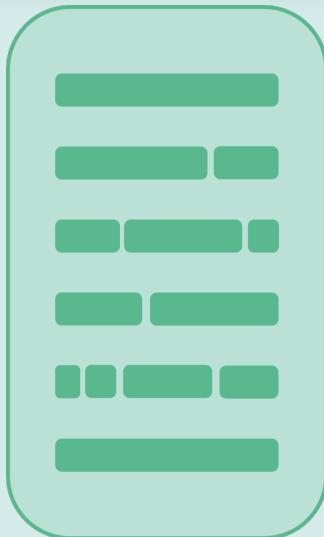
```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class HelloWorldController {

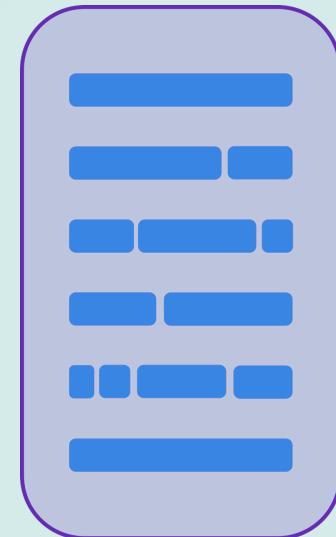
    @GetMapping(value = "/hello-world",
                headers = "HX-Request")
    String helloWorldResponse() {
        return "hello-world-response";
    }
}
```

"Fullstack"-Anwendungen (Meta-Frameworks)

"Fullstack"-Anwendungen (Meta-Frameworks)

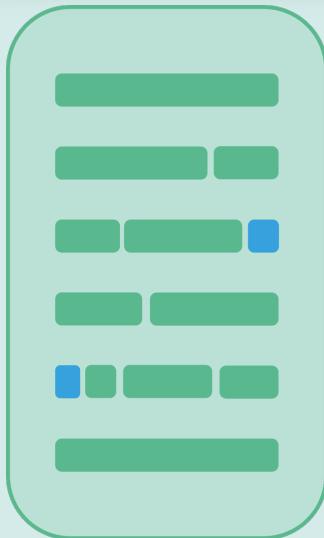


Nur Server (+JS)

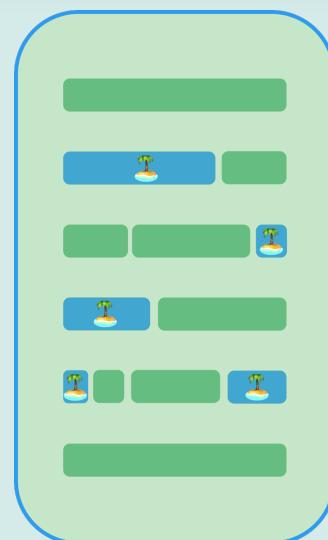


Nur Client (SPA)

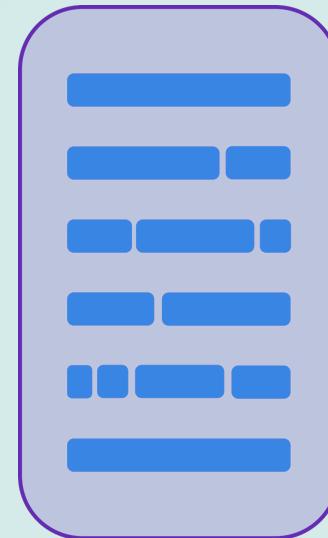
"Fullstack"-Anwendungen (Meta-Frameworks)



Nur Server (+JS)



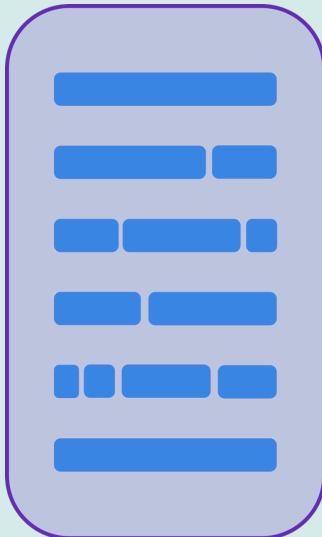
JS Fullstack Anwendung



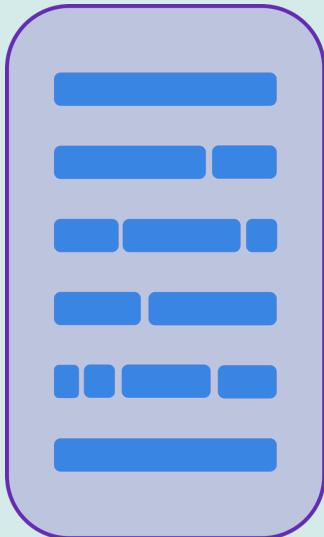
Nur Client (SPA)

FULLSTACK-ANWENDUNGEN (JAVASCRIPT)

Fullstack-Anwendungen (JavaScript)

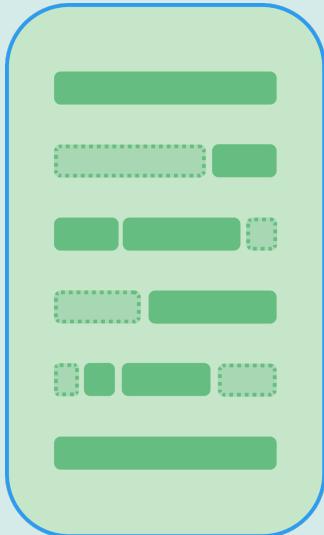


Fullstack-Anwendungen (JavaScript)



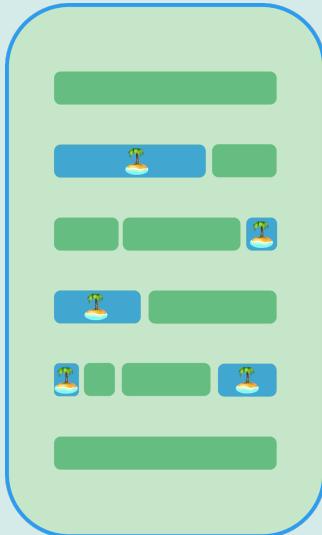
- Ebenfalls vollständig in **JavaScript** geschrieben

Fullstack-Anwendungen (JavaScript)



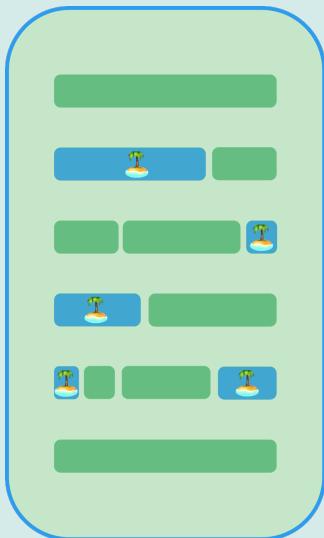
- Grundsätzliche Idee:
 1. **UI-Code** wird serverseitig vorgerendert
 2. **UI-Code** wird zum Browser gesendet und angezeigt

Fullstack-Anwendungen (JavaScript)



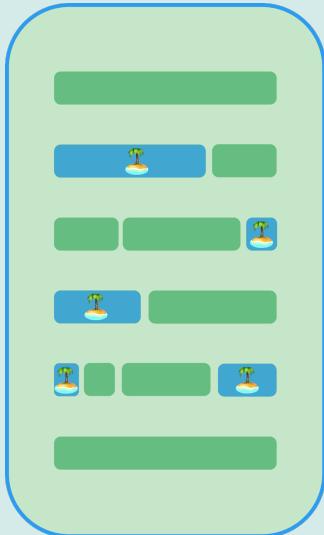
- Grundsätzliche Idee:
 1. **UI-Code** wird serverseitig vorgerendert
 2. **UI-Code** wird zum Browser gesendet und angezeigt
 3. Nur der JavaScript-Code ("Islands") **für Interaktionen** wird zum Browser geschickt

Fullstack-Anwendungen (JavaScript)



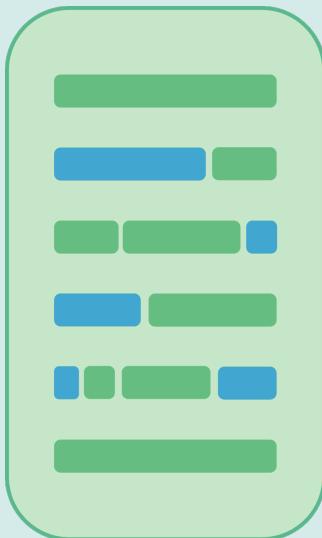
- Bekannte Vertreter:
 1. **Next.js** (React)
 2. SvelteKit (Svelte)
 3. Nuxt.js (Vue)
 4. Astro (eigenes Framework + Support für alle SPAs)
 5. Qwik (eigenes Framework)

Fullstack-Anwendungen (JavaScript)



- Bekannte Vertreter:
 1. **Next.js** (React)
 2. SvelteKit (Svelte)
 3. Nuxt.js (Vue)
 4. Astro (eigenes Framework + Support für alle SPAs)
 5. Qwik (eigenes Framework)
- Funktionalität und Herangehensweise unterschiedlich

Fullstack-Anwendungen (JavaScript)



- Wir sind zurück zur **JavaScript-Schnipsel**-Architektur
 - aber: die Schnipsel werden **automatisch** vom Framework erzeugt
 - die Schnipsel existieren nur zur **Laufzeit**

Fullstack

oder

htmx



BEISPIEL-ANWENDUNG

- Eine Beispiel-Anwendung...
- <http://localhost:9090>



Code-Beispiele im Editor!

HTMX-Anwendung: <http://localhost:9080>

CODE-BEISPIELE



Code-Beispiele im Editor!

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

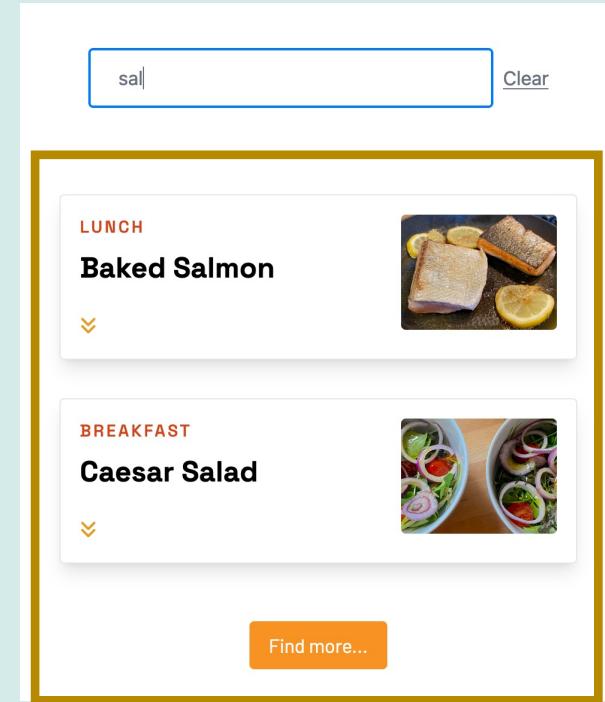
 [Clear](#)

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

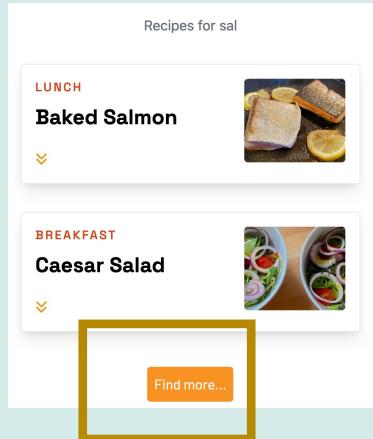
 [Clear](#)

GET /search



CODE-BEISPIELE

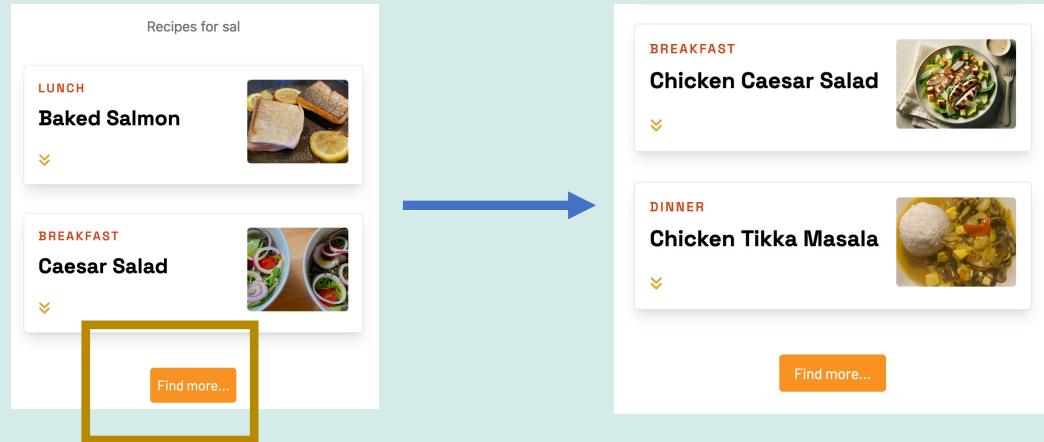
Beispiel: Bestehende Darstellung aktualisieren



GET /search?page=2

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren



GET /search?page=2

hx-swap="beforeend"

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH

Baked Salmon

▼



[Find more...](#)

BREAKFAST

Caesar Salad

▼



HTMX Request

BREAKFAST

Chicken Caesar Salad

▼



DINNER

Chicken Tikka Masala

▼



[Find more...](#)

GET /search?page=2

hx-swap="beforeend"

Recipes for sal

LUNCH

Baked Salmon

▼



[Find more...](#)

BREAKFAST

Caesar Salad

▼



[Find more...](#)

BREAKFAST

Chicken Caesar Salad

▼



DINNER

Chicken Tikka Masala

▼



[Find more...](#)

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH
Baked Salmon

BREAKFAST
Caesar Salad

[Find more...](#)

HTMX Request

BREAKFAST
Chicken Caesar Salad

DINNER
Chicken Tikka Masala

[Find more...](#)

GET /search?page=2

hx-swap="beforeend"



Recipes for sal

LUNCH
Baked Salmon

BREAKFAST
Caesar Salad

[Find more...](#)

BREAKFAST
Chicken Caesar Salad

DINNER
Chicken Tikka Masala

[Find more...](#)

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

[Find more...](#)

HTMX Request

Recipes for sal

BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

[Find more...](#)

GET /search?page=2

hx-swap="beforeend"



Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

[Find more...](#)

GET /search?page=2

BREAKFAST
Chicken Caesar Salad
▼ 

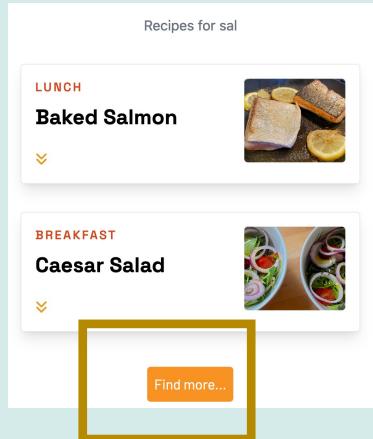
DINNER
Chicken Tikka Masala
▼ 

[Find more...](#)

GET /search?page=3

CODE-BEISPIELE

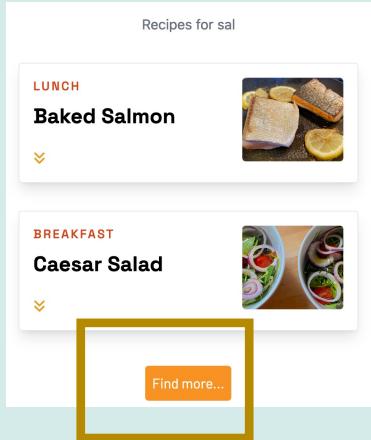
Beispiel: Bestehende Darstellung aktualisieren



GET /search?page=2

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren



Find more...

```
id="moreButton"  
hx-swap-oob="true"
```

GET /search?page=2

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH

Baked Salmon

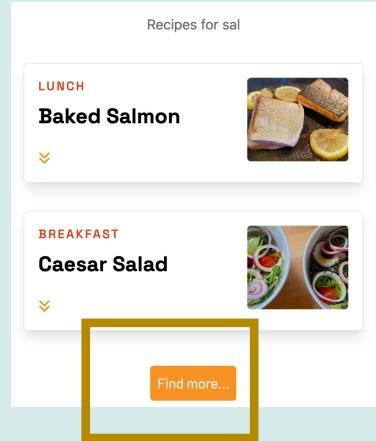
▼

BREAKFAST

Caesar Salad

▼

[Find more...](#)



HTMX Request

[Find more...](#)

`id="moreButton"`

`hx-swap-oob="true"`

BREAKFAST

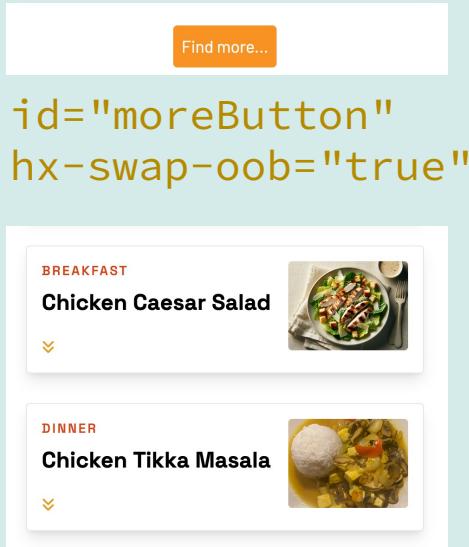
Chicken Caesar Salad

▼

DINNER

Chicken Tikka Masala

▼



`hx-target="result"`

`hx-swap="beforeend"`

GET /search?page=2

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

[Find more...](#)

HTMX Request

`id="moreButton"
hx-swap-oob="true"`

BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

`hx-target="result"
hx-swap="beforeend"`



Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

[Find more...](#)

GET /search?page=2

GET /search?page=3

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

Find more...

HTMX Request

GET /search?page=2

id="moreButton"
hx-swap-oob="true"

BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

hx-target="result"
hx-swap="beforeend"



Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

Find more...

GET /search?page=3

CODE-BEISPIELE

Beispiel: Bestehende Darstellung aktualisieren

Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

[Find more...](#)

GET /search?page=2

FindMoreButton.jte

RecipeSearch.tsx

Find more...

`id="moreButton"
hx-swap-oob="true"`

BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

`hx-target="result"
hx-swap="beforeend"`



Find more...

Recipes for sal

LUNCH
Baked Salmon
▼ 

BREAKFAST
Caesar Salad
▼ 

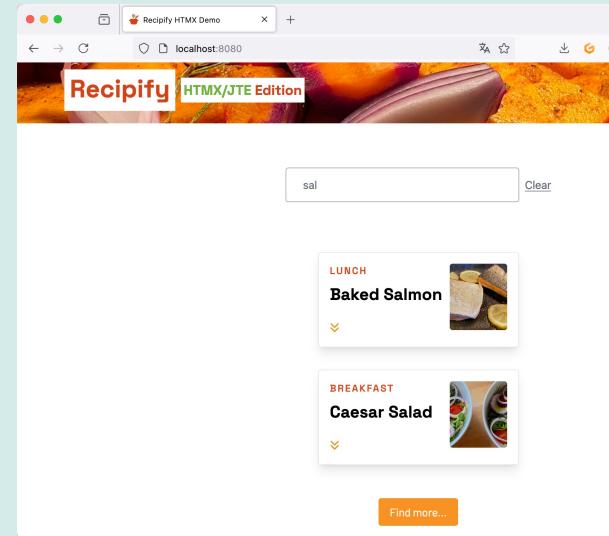
BREAKFAST
Chicken Caesar Salad
▼ 

DINNER
Chicken Tikka Masala
▼ 

GET /search?page=3

Beispiel: Parallel Requests

-  Netzwerk-Tab

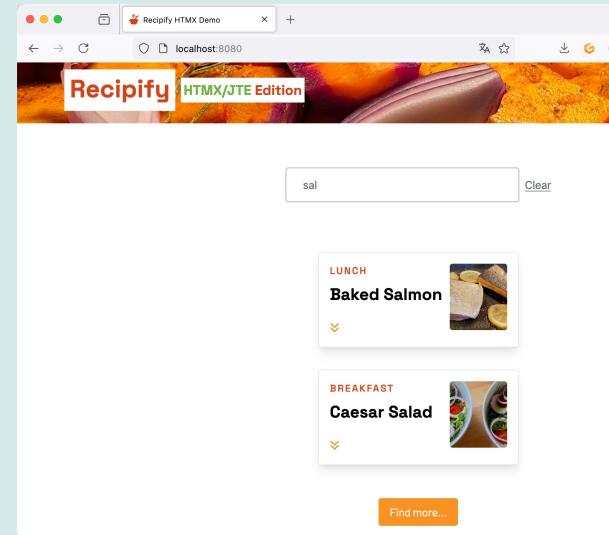


```
SearchPageContent.jte: slowdown=2000, dafür delay raus
```

```
demo-config: slowDown_search=2000, debounce_search=0
```

Beispiel: Parallel Requests

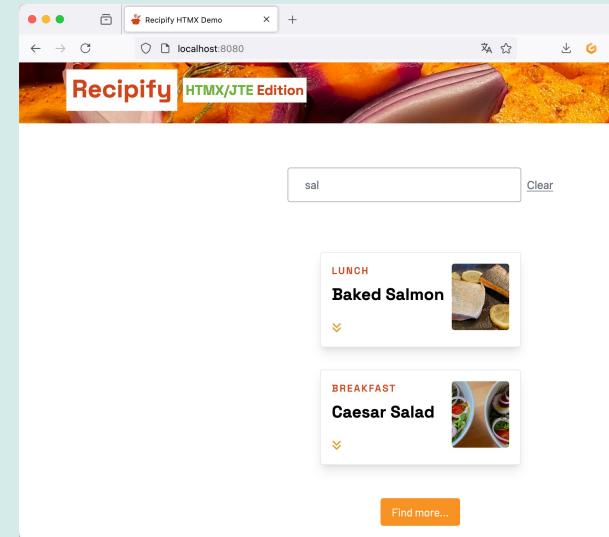
- 🕵️ Netzwerk-Tab
- HTMX führt Requests nacheinander aus
- In der SPA können wir das selbst entscheiden



Beispiel: Parallel Requests

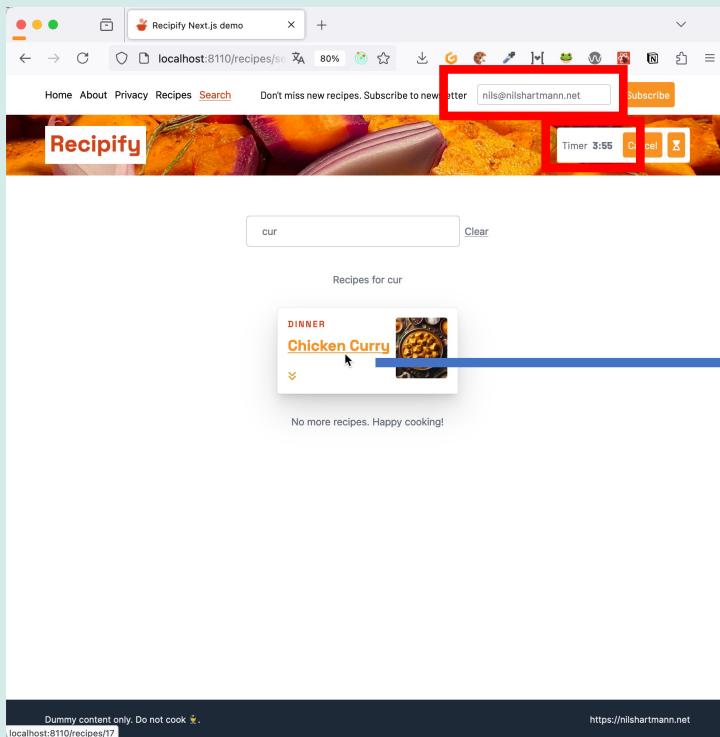
- 🕵️ Netzwerk-Tab
- HTMX führt Requests nacheinander aus
- "Alte" Request werden nicht verworfen
- In der SPA können wir das selbst entscheiden

SearchPageContent.jte: slowdown

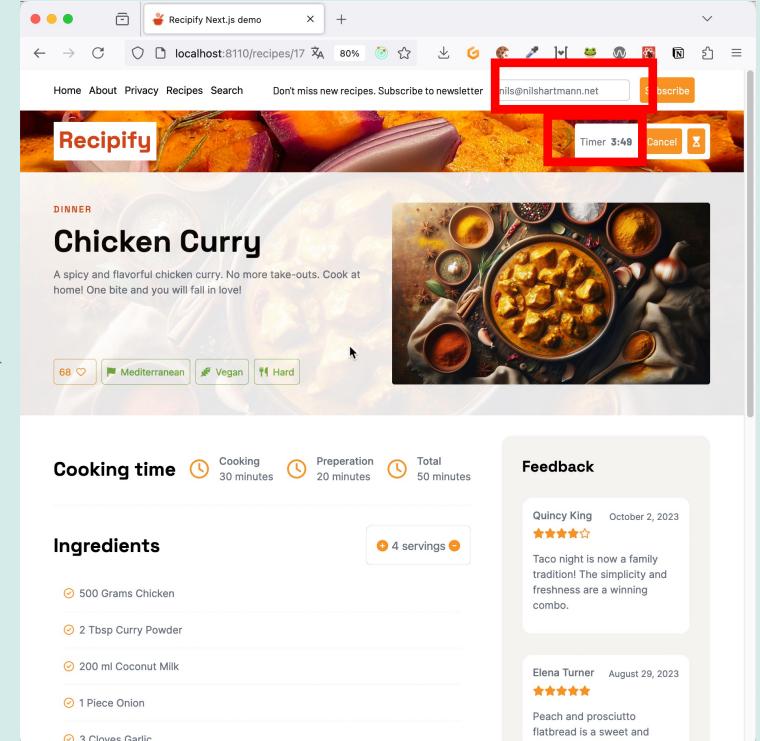


Beispiel: SPA-like Seitenwechsel

- Beim Aufruf der Rezept-Seite soll der Rahmen unverändert bleiben



/recipes/17



Beispiel: SPA-like Seitenwechsel in Next.js

- **Link** verhält sich automatisch richtig:

- mit und ohne JS
- Ganze Seite vs. nur Content
- Aktualisieren der URL

```
<Link
  href={`/recipes/${recipe.id}`}
>
  {recipe.title}
</Link>
```

(RecipeSummaryCard.tsx)

Beispiel: SPA-like Seitenwechsel in Next.js

- Link verhält sich automatisch richtig:

- mit und ohne JS
- Ganze Seite vs. nur Content
- Aktualisieren der URL

```
<Link  
  href={`/recipes/${recipe.id}`}  
>  
  {recipe.title}  
</Link>
```

(RecipeSummaryCard.tsx)

- "Controller": Identischer Code
 - Fullpage Request
 - Incremental Update

[recipeId]/page.tsx

recipes/layout.tsx

Beispiel: SPA-like Seitenwechsel in Next.js

- Link verhält sich automatisch richtig:

- mit und ohne JS
- Ganze Seite vs. nur Content
- Aktualisieren der URL

```
<Link  
  href={`/recipes/${recipe.id}`}  
>  
  {recipe.title}  
</Link>
```

(RecipeSummaryCard.tsx)

- "Controller": Identischer Code

- Fullpage Request
- Incremental Update

[recipeId]/page.tsx

recipes/layout.tsx

- Das funktioniert **clientseitig** mit allen SPA-Routern

Beispiel: SPA-like Seitenwechsel in HTMX

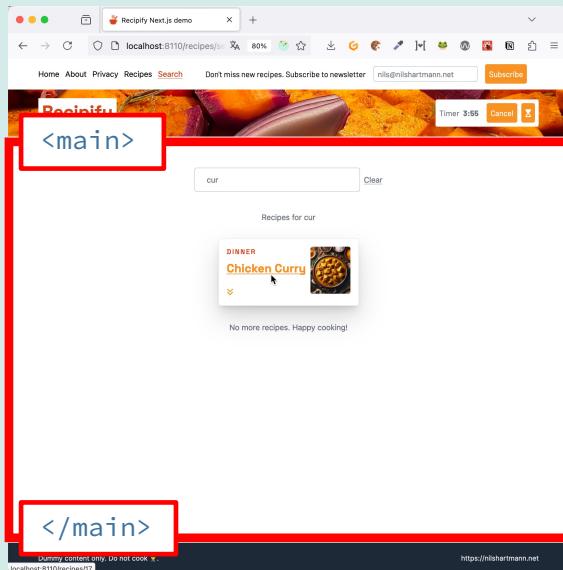
- Link muss entsprechend ausgezeichnet werden

- Für Klick ohne JavaScript
- Für Klick mit JavaScript
- Aktualisieren der URL im Browser

RecipeSummaryCard.jte

Beispiel: SPA-like Seitenwechsel in HTMX

Variante 1



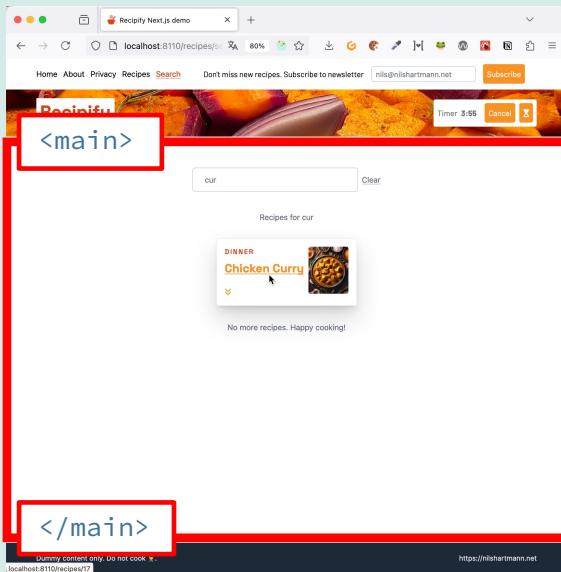
RecipeSummaryCard.jte

CODE-BEISPIELE

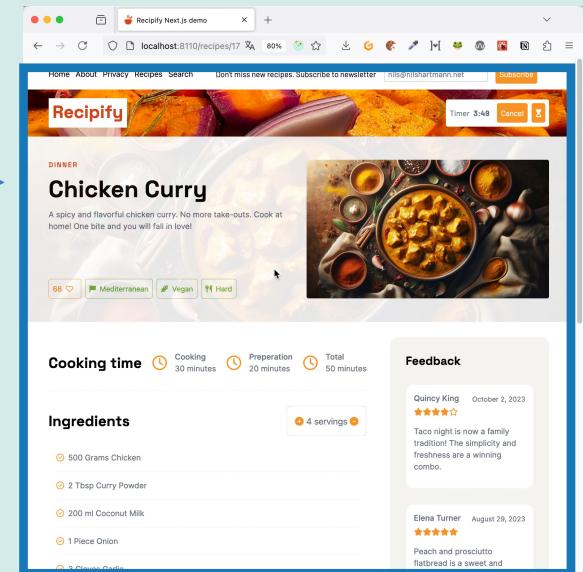
Beispiel: SPA-like Seitenwechsel in HTMX

Variante 1

RecipeSummaryCard.jte



/recipes/17

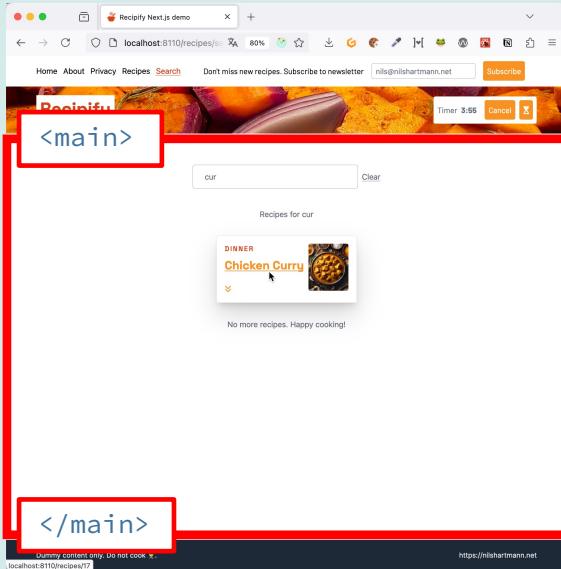


CODE-BEISPIELE

Beispiel: SPA-like Seitenwechsel in HTMX

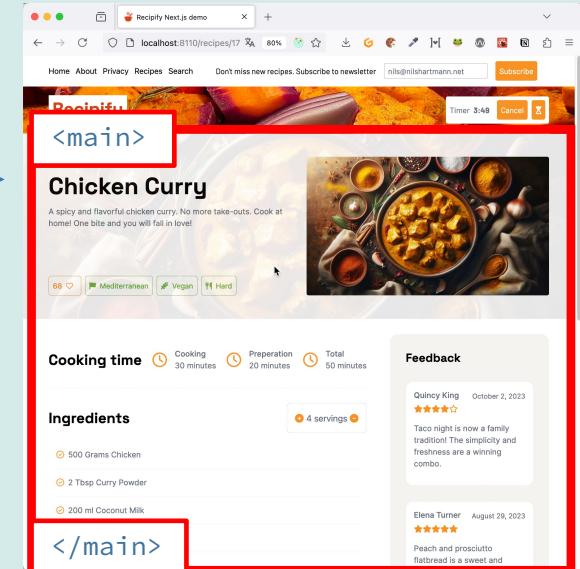
Variante 1

RecipeSummaryCard.jte



/recipes/17

hx-select="main"

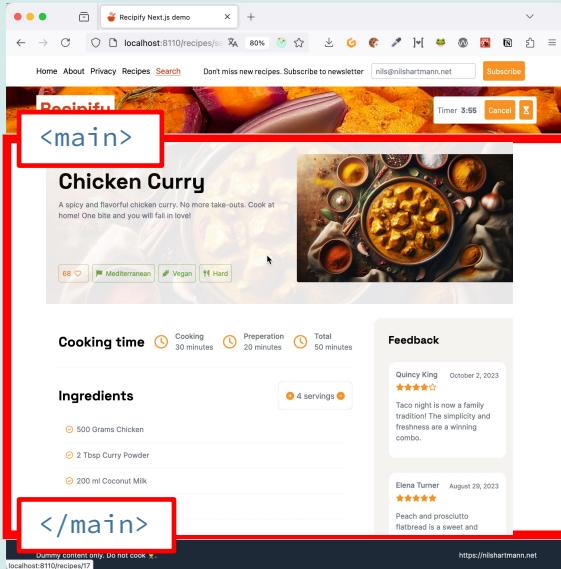


CODE-BEISPIELE

Beispiel: SPA-like Seitenwechsel in HTMX

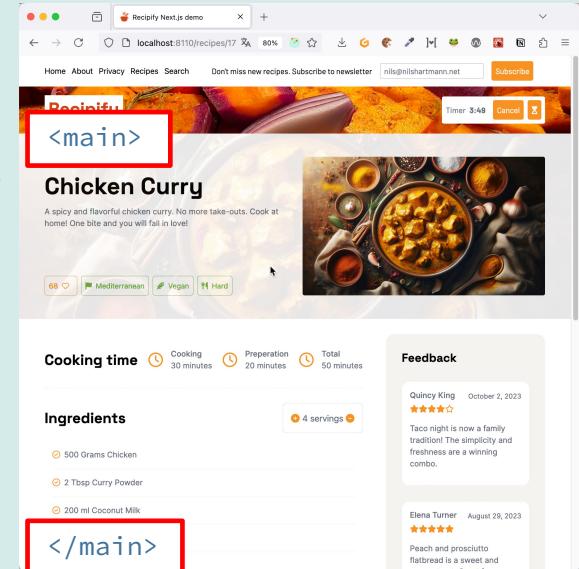
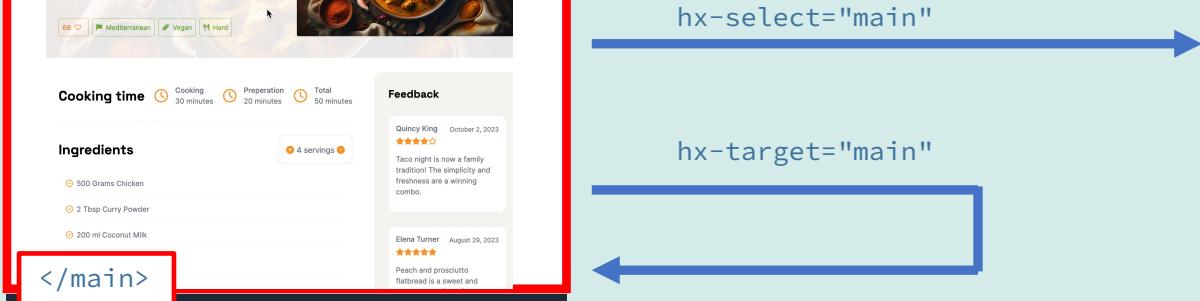
Variante 1

RecipeSummaryCard.jte



/recipes/17

hx-select="main"



</main>

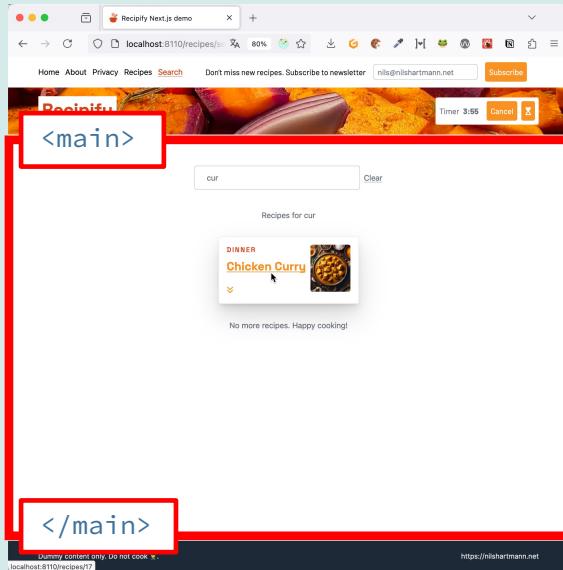
Variante 1: HTMX

- **Link** muss wissen:

- wie sieht das Ergebnis aus
- welches Element brauche ich daraus
- wo muss das auf der aktuellen Seite eingebaut werden

Beispiel: SPA-like Seitenwechsel in HTMX

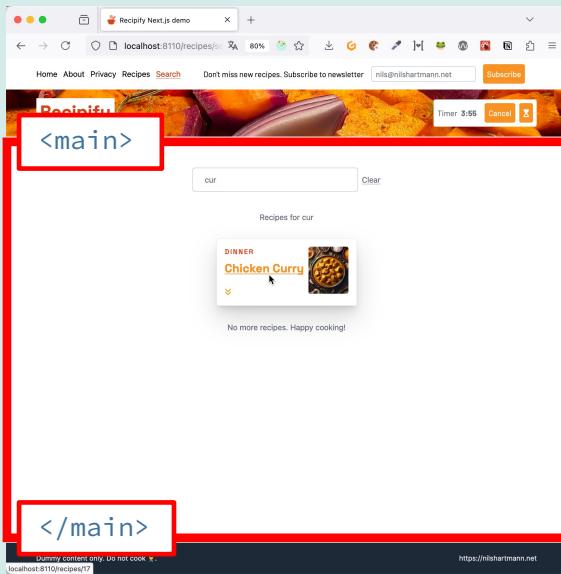
Variante 2: Eigener Endpunkt



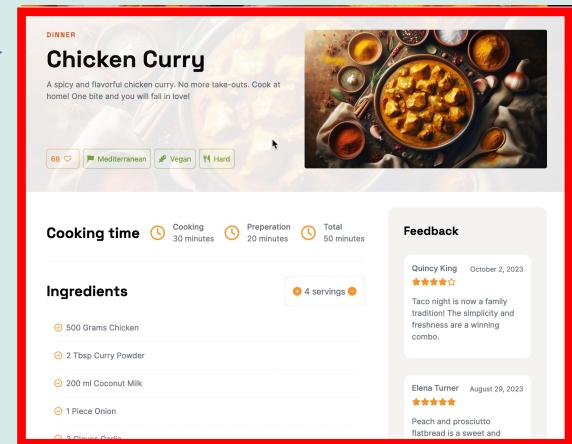
CODE-BEISPIELE

Beispiel: SPA-like Seitenwechsel in HTMX

Variante 2: Eigener Endpunkt



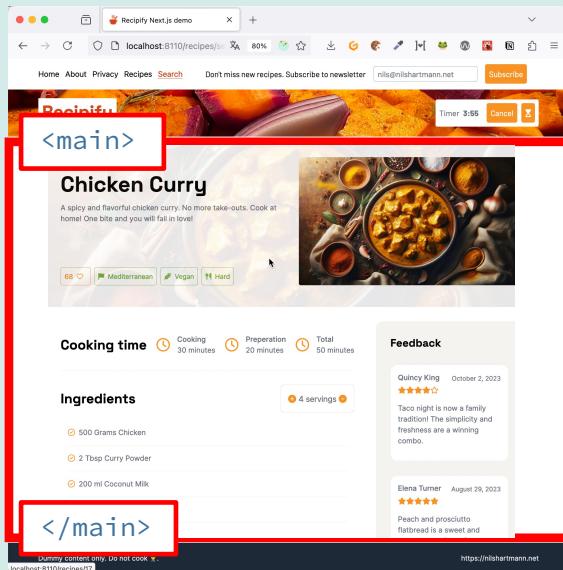
/recipes/17/details



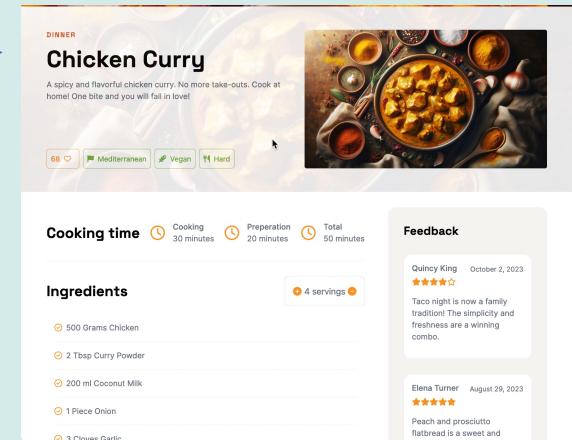
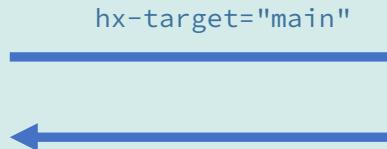
CODE-BEISPIELE

Beispiel: SPA-like Seitenwechsel in HTMX

Variante 2



/recipes/17/details



Variante 2: Eigener Endpunkt

- **Link** muss wissen:

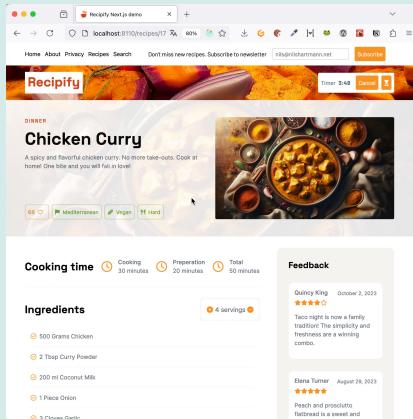
- welcher Endpunkt
- wo muss Ergebnis auf der aktuellen Seite eingebaut werden

Variante 2: Eigener Endpunkt

- **Backend** muss zwei Endpunkte zur Verfügung stellen:

Variante 2: Eigener Endpunkt

- **Backend** muss zwei Endpunkte zur Verfügung stellen:
 - Ganze Seite



`/recipes/17`

Variante 2: Eigener Endpunkt

- Backend muss zwei Endpunkte zur Verfügung stellen:
 - Ganze Seite
 - Nur den Mittelteil

The screenshot shows a web browser window with the title 'Recipify Nextjs demo'. The main content is a recipe for 'Chicken Curry'. At the top, there's a navigation bar with links for Home, About, Privacy, Recipes, Search, and a newsletter sign-up. Below the navigation is a banner for 'Recipify' with a photo of a dish. The main section has a heading 'DINNER' and 'Chicken Curry'. A sub-headline says 'A spicy and flavorful chicken curry for more take-outs. Cook at home! Once bite and you will fall in love!' Below this is a large image of the dish. To the right of the image are buttons for '68', 'Mediterranean', 'Vegan', and 'Hard'. Under the heading, there are sections for 'Cooking time' (30 minutes), 'Preparation' (20 minutes), and 'Total' (50 minutes). A 'Feedback' section contains two reviews: one from 'Quincy King' (Oct 02, 2023) and one from 'Elana Turner' (Aug 29, 2023). At the bottom, there's a 'Ingredients' section listing: 500 Grams Chicken, 2 Tbsp Curry Powder, 200 ml Coconut Milk, 1 Peice Onion, and 3 Cloves Garlic. It also indicates '4 servings'.

/recipes/17

This screenshot shows the same recipe page but with a different URL: '/recipes/17/details'. The layout is identical to the first screenshot, featuring the 'Recipify' banner, the 'Chicken Curry' recipe card with its details, cooking times, and the feedback section with reviews from Quincy King and Elana Turner. The ingredients list at the bottom is identical to the first screenshot.

/recipes/17/details

CODE-BEISPIELE

Beispiel: Seite priorisieren

The screenshot shows a recipe card for "Classic Caesar Salad" on the Recipify platform. The card includes the following details:

- Category:** BREAKFAST
- Title:** Classic Caesar Salad
- Description:** Crispy romaine lettuce with creamy Caesar dressing. Bon Appétit! Food that feels like home.
- Image:** A large, vibrant photograph of a Caesar salad in a bowl, garnished with croutons and parmesan cheese.
- Metrics:** 92 likes, Asian cuisine, Vegan, Easy.
- Cooking time:** Cooking 10 minutes, Preparation 10 minutes, Total 10 minutes.
- Ingredients:** 1 Head Romaine Lettuce, 100 ml Caesar Dressing, 100 Grams Croutons, 50 Grams Parmesan.
- Instructions:** (Listed below the ingredients)
- Feedback:** Loading feedback... (with three orange dots).
- Your opinion?** (Form fields for name and rating).

CODE-BEISPIELE

Beispiel: Seite priorisieren in Next.js

慢速请求反馈慢速请求反馈
slowDown_GetFeedbacks=3000

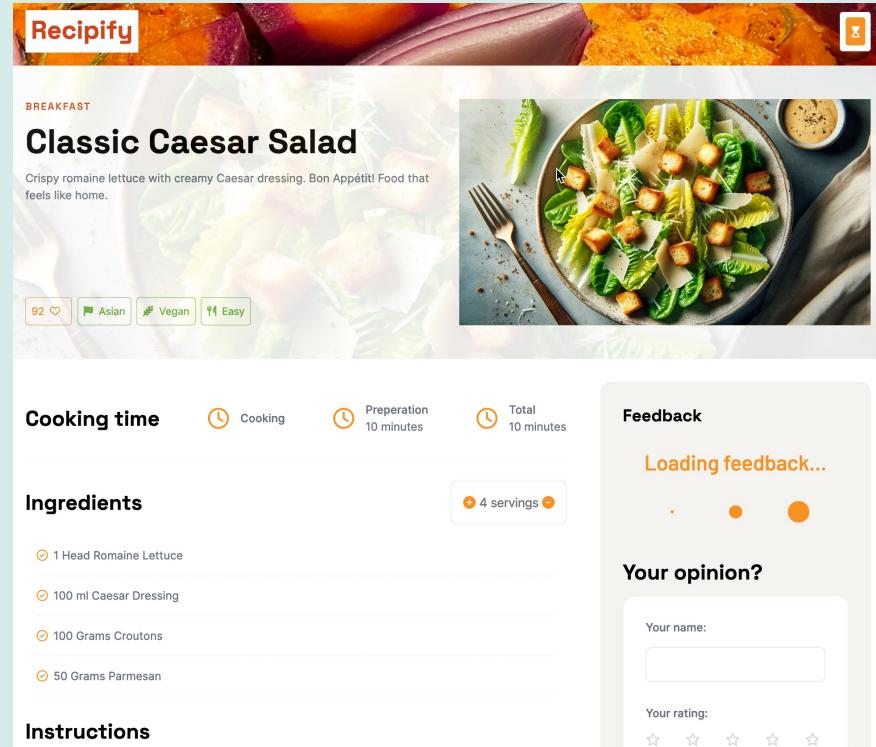
慢速请求慢速请求
slowDown_GetRecipe=2000

慢速请求慢速请求
Spring Controller Debug

慢速请求慢速请求
Network-Tab Total Duration

慢速请求慢速请求
[recipeId]/page.tsx

慢速请求慢速请求
RecipePageContent.tsx



CODE-BEISPIELE

Beispiel: Seite priorisieren in HTMX

The screenshot displays a cooking recipe page for "Classic Caesar Salad" on the Recipify website. The page is structured to prioritize content through HTMX, allowing parts of the page to update without reloading.

Header: Recipify logo and a close button (X).

Section Headers: BREAKFAST, Classic Caesar Salad.

Description: Crispy romaine lettuce with creamy Caesar dressing. Bon Appétit! Food that feels like home.

Meta Data: 92 likes, Asian, Vegan, Easy.

Cooking Time: Cooking time (clock icon), Preparation 10 minutes, Total 10 minutes.

Feedback: Loading feedback...

Ingredients: 1 Head Romaine Lettuce, 100 ml Caesar Dressing, 100 Grams Croutons, 50 Grams Parmesan. A note indicates 4 servings.

Instructions:

Form Fields: Your name: (input field), Your rating: (rating scale from 1 to 5 stars).

CODE-BEISPIELE

Beispiel: Seite priorisieren in HTMX



Feedback.jte



<http://localhost:8080/recipes/25?slowdown=2000>



Netzwerk-Tab

The screenshot shows a recipe card for "Classic Caesar Salad" from the "BREKFAST" section. The card includes a thumbnail image of the salad, cooking time (10 minutes), preparation time (10 minutes), ingredients (Romaine Lettuce, Caesar Dressing, Croutons, Parmesan), instructions, and a feedback section with a rating of 3 stars.

Recipify

BREAKFAST

Classic Caesar Salad

Crispy romaine lettuce with creamy Caesar dressing. Bon Appétit! Food that feels like home.

92 ❤️ Asian Vegan Easy

Cooking time Cooking 10 minutes Preparation 10 minutes Total 10 minutes

Ingredients

- 1 Head Romaine Lettuce
- 100 ml Caesar Dressing
- 100 Grams Croutons
- 50 Grams Parmesan

Instructions

Feedback

Loading feedback...

Your opinion?

Your name:

Your rating:

Beispiel: Clientseitige Interaktion

- Clientseitige Interaktion

Ingredients

+ 4 servings -

✓ 1.0 Piece Chicken Breast

✓ 1.0 Head Romaine Lettuce

✓ 100.0 ml Caesar Dressing

✓ 100.0 Grams Croutons

✓ 50.0 Grams Parmesan

Beispiel: Clientseitige Interaktion in Next.js



IngredientsSection.tsx

Ingredients

+ 4 servings -

✓ 1.0 Piece Chicken Breast

✓ 1.0 Head Romaine Lettuce

✓ 100.0 ml Caesar Dressing

✓ 100.0 Grams Croutons

✓ 50.0 Grams Parmesan

Beispiel: Clientseitige Interaktion in HTMX

- Wir brauchen spätestens hier JavaScript



Ingredients.jte

Ingredients

+ 4 servings -

✓ 1.0 Piece Chicken Breast

✓ 1.0 Head Romaine Lettuce

✓ 100.0 ml Caesar Dressing

✓ 100.0 Grams Croutons

✓ 50.0 Grams Parmesan

Beispiel: Clientseitige Interaktion in HTMX

- Wir brauchen spätestens hier JavaScript



SearchPageContent.jte

The screenshot shows a search interface. At the top right is a small icon of a person wearing a hat and coat. To its right is the text "SearchPageContent.jte". Below this is a search bar with a placeholder text "Type three letters to start search" enclosed in a yellow rectangular border. To the right of the search bar is a "Clear" button.

Webanwendungendungen

Fazit

SPA

oder

htmx



SPA/Fullstack oder HTMX

- It depends (natürlich 😐)

SPA/Fullstack oder HTMX

- It depends (natürlich 😐)
- Will ich eine "Hypertext"-Anwendung bauen
oder
- eine Anwendung, die die Anforderungen meiner Benutzer:innen erfüllt? 😊

SPA/Fullstack oder HTMX

- Für mich ist "wir machen nur HTML plus bisschen JS" zu kurz gesprungen
- Das sieht man auch an der Vielzahl der HTMX-Attribute
- Wenn man einen Server-Stack schon hat, vielleicht legitime Wahl

Executive Summary

- The effort took about 2 months (with a 21K LOC code base, mostly JavaScript)
- No reduction in the application's user experience (UX)
- They reduced the code base size by 67% (21,500 LOC to 7200 LOC)
- They **increased python code by 140%** (500 LOC to 1200 LOC), a good thing if you prefer python to JS
- They reduced their total JS dependencies by 96% (255 to 9)
- They reduced their web build time by 88% (40 seconds to 5)

Executive Summary

These:

Wenn "if you prefer YOUR-BACKEND-LANGUAGE-HERE to JS" zutrifft,

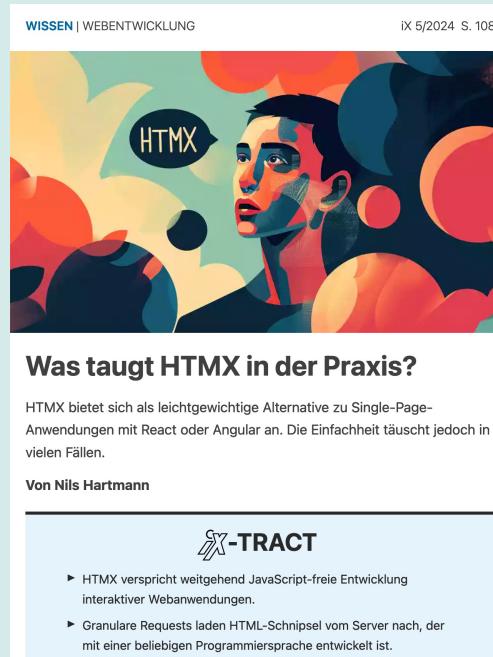
dann ist man vielleicht falsch in der Frontend-Entwicklung



reduced python code by 140% (500 LOC to 1200 LOC), a good thing if
you prefer python to JS

- They reduced their total JS dependencies by 96% (255 to 9)
- They reduced their web build time by 88% (40 seconds to 5)

Mein Artikel (heise / iX) zum Thema: <https://react.schule/heise-htmx>



The screenshot shows a magazine page from 'WISSEN | WEBENTWICKLUNG' issue 'iX 5/2024 S. 108'. The main image is a colorful abstract illustration of a person's face surrounded by large, stylized, overlapping shapes in shades of red, orange, yellow, and blue. A speech bubble above the head contains the text 'HTMX'. Below the image, the title 'Was taugt HTMX in der Praxis?' is displayed in bold black font. A short text summary follows: 'HTMX bietet sich als leichtgewichtige Alternative zu Single-Page-Anwendungen mit React oder Angular an. Die Einfachheit täuscht jedoch in vielen Fällen.' The author's name, 'Von Nils Hartmann', is listed below the summary. At the bottom, there is a section titled 'iX-TRACT' with two bullet points: 'HTMX verspricht weitgehend JavaScript-freie Entwicklung interaktiver Webanwendungen.' and 'Granulare Requests laden HTML-Schnipsel vom Server nach, der mit einer beliebigen Programmiersprache entwickelt ist.'

Vielen Dank!

Code & Slides: <https://react.schule/md-dev-days-htmxt>

Fragen und Kontakt:

nils@nilshartmann.net

<https://nilshartmann.net/kontakt>

