

NILS HARTMANN

<https://nilshartmann.net>

Frontend-Entwicklung

Single-

Page-

Anwendung

oder

HTMX?

NILS HARTMANN

nils@nilshartmann.net

Freiberuflischer Entwickler, Architekt, Trainer

Java, Spring, GraphQL, React, TypeScript



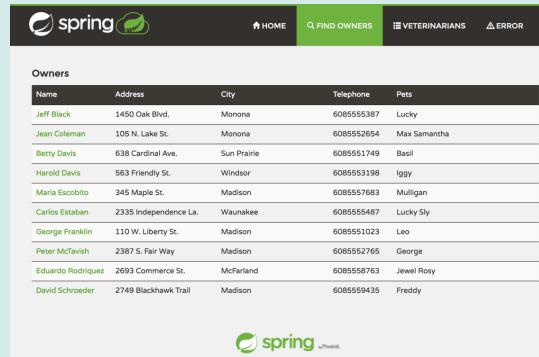
<https://graphql.schule/video-kurs>

<https://reactbuch.de>

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

Webanwendungen...

WEBANWENDUNGEN



The screenshot displays a web application interface for managing pet owners. At the top, there is a navigation bar with the Spring logo, followed by links for HOME, FIND OWNERS (highlighted in green), VETERINARIANS, and ERROR.

The main content area is titled "Owners" and contains a table with the following data:

Name	Address	City	Telephone	Pets
Jeff Black	1450 Oak Blvd.	Monona	6085555387	Lucky
Jean Coleman	105 N. Lake St.	Monona	6085552654	Mox Samantha
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085511749	Basil
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Maria Escobito	345 Maple St.	Madison	608557683	Mulligan
Carlos Esteban	2335 Independence La.	Wauakee	6085555487	Lucky Sly
George Franklin	110 W. Liberty St.	Madison	6085511023	Leo
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	Jewel Rosy
David Schroeder	2749 Blackhawk Trail	Madison	6085559435	Freddy

At the bottom of the page, there is a footer with the Spring logo and the word "powered".

WEBANWENDUNGEN

The screenshot shows a web browser window titled "ICE Portal" displaying travel information for ICE 109. At the top, it says "Reiseinformationen für den ICE 109". It indicates the next stop in 6 minutes at Gleis 2. The train is currently at Osnabrück Hbf, shown as "108 km/h" and "Schnelles Internet". The travel time is listed as "12:35 12:46". Below this, a section titled "Ihre Reise in Richtung Basel SBB" shows the route from Hamburg-Altona to Gl. 10. It highlights the connection "ICE 109 nach Basel SBB" with 14 stops hidden. Buttons for "Vergangene Halte zeigen", "Ab", "An", "Suchen", "Erweiterte Suche", and "Aktuelle Meldungen" are visible. The bottom of the page features promotional banners for "hw plus" and "hvv Deutschlandticket", along with social media links for "Mach mit!" and "Schüler*innen".

The screenshot shows a web application titled "spring". The main navigation bar includes "HOME", "FIND OWNERS" (which is highlighted in green), "VETERINARIANS", and "ERROR". Below the navigation, a table lists "Owners".

Name	Address	City	Telephone	Pets
Jeff Black	1450 Oak Blvd.	Monona	6085555387	Lucky
Jean Coleman	105 N. Lake St.	Monona	6085552654	Max Samantha
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085511749	Basil
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Maria Escobito	345 Maple St.	Madison	608557683	Mulligan
Carlos Esteban	2335 Independence La.	Wauakee	6085555487	Lucky Sly
George Franklin	110 W. Liberty St.	Madison	6085511023	Leo
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	Jewel Rosy
David Schroeder	2749 Blackhawk Trail	Madison	6085559435	Freddy

At the bottom, there is a logo for "spring" and the word "invest".

WEBANWENDUNGEN

The image displays four distinct web applications arranged horizontally:

- Booking.com**: A travel booking website showing flight information for an ICE 109 train. It includes details like the next stop at Osnabrück Hbf (Gleis 2) in 6 minutes, current speed (108 km/h), and a Genius Prämien offer for Nils.
- ICE Portal**: A real-time train status application for the ICE 109. It shows the train's current location at Hamburg-Altona and its destination at Basel SBB, with 14 stops hidden. It also displays a speed of 108 km/h and a connection to Schnelles Internet.
- Reiseinformationen für den ICE 109**: A separate window or tab showing detailed travel information for the ICE 109, including arrival times at various stops along the route.
- Spring**: A platform for pet owners. The interface includes a navigation bar with HOME, FIND OWNERS, VETERINARIANS, and ERROR links. The main content area displays a table of pet owners with columns for Name, Address, City, Telephone, and Pets. The table lists 14 entries, such as Jeff Black (1450 Oak Blvd., Monona, 608555387, Lucky) and Betty Davis (638 Cardinal Ave., Sun Prairie, 608551749, Basil).

WEBANWENDUNGEN

The image displays a Mac desktop environment with several open windows illustrating various web applications:

- Booking.com**: A travel search engine showing flight information for the ICE 109 from Osnabrück Hbf to Basel SBB.
- ICE Portal**: A real-time train status application showing the ICE 109 at 108 km/h on Gleis 2, with the next stop in 6 minutes.
- Hibernate ORM**: A screenshot of a Jira board titled "Hibernate" showing a list of tasks under the "Vorgänge" (Tasks) section.
- sessionize**: A speaker dashboard for Nils Hartmann, showing a public profile and session details.
- g-graphql-training**: A GitHub repository page for a GraphQL training project, showing pull requests and code snippets.
- Attendee list for Frontend for Backend: HTMX oder Single-Page-Anwendung?**: An event attendee list from Ichiba GmbH, showing 80 attendees.

WEBANWENDUNGEN

The image displays a collage of several web application interfaces:

- Booking.com**: A travel booking website showing flight information for the ICE 109 from Osnabrück Hbf to Basel SBB.
- ICE Portal**: A real-time train status board showing the next stop at Gleis 2 in 6 minutes, speed (108 km/h), and a link to 'Schnelles Internet'.
- Hibernate**: A project management tool showing a list of tasks and projects related to 'Hibernate ORM'.
- Microsoft Teams**: A communication platform showing a chat history with Nils Hartmann and other team members.
- Sessionize**: An event management tool showing a speaker dashboard for Nils Hartmann.
- GitHub**: A code repository for a 'spring-graphql-training' project, listing pull requests and issues.
- Eventbrite**: A screenshot of an event page for a 'Frontend for Backend: HTMX oder Single-Page-Anwendung?' event.

WEBANWENDUNGEN

The image displays a collage of several web application screenshots, illustrating various types of web-based tools and services:

- Booking.com**: A travel booking website showing flight information for a flight from Osnabrück Hbf to Nürnberg (Nes, Nils) at 12:46.
- ICE Portal**: A real-time train status application showing the next stop for ICE 109 at Gleis 2 in 6 minutes, with a speed of 108 km/h.
- ChatGPT - DALLE**: A collaboration between AI models. ChatGPT asks if a user can generate a teaser image for a cooking recipe website for "Classic Beef Burger". DALLE responds with two images of a beef burger.
- HIBERNATE**: A project management tool for the Hibernate ORM software project, showing a list of tasks and their status.
- sessionize**: A speaker dashboard for Nils Hartmann, showing session details and a QR code.
- spring**: A developer-oriented application showing a list of owners with columns for Name, Address, City, Telephone, and Pets.
- Microsoft Teams**: A communication platform showing a chat history with messages from Florian Riechers and others.

WEBANWENDUNGEN

Die Webanwendung gibt es nicht

The image is a collage of several screenshots from various web applications, demonstrating a wide range of software tools and services:

- ICE Portal:** A travel-related portal showing a map of Europe with flight routes and a search bar.
- Booking.com:** A screenshot of the Booking.com homepage.
- ChatGPT - DALL-E:** A conversation with AI models. One message shows two images of a beef burger, with the caption: "Here are two teaser images for a cooking recipe website featuring a 'Classic Beef Burger'. These images are designed to appeal to a young, modern urban audience."
- Hibernate ORM:** A screenshot of a GitHub repository for the Hibernate ORM project, showing the "Vorgänge" (Tasks) page.
- sessionize:** A screenshot of the sessionize app interface, showing a user profile for Nils Hartmann.
- spring:** A screenshot of the spring framework website, showing the "Owners" section of a database table.
- Google Sheets:** A screenshot of a Google Sheets spreadsheet with multiple tabs and data entries.
- Microsoft Teams:** A screenshot of a Microsoft Teams chat window with a message about a GitHub pull request.

WEBANWENDUNGEN

Die Webanwendung gibt es nicht
...aber: (fast) alle brauchen JavaScript

Booking.com

Hibernate ORM Softwareprojekt

Vorgänge

DALL-E

hvv plus

hvv Deutschlandticket

Mach mit!

spring

Owners

Name	Address	City	Telephone	Pets
Jeff Black	1450 Oak Blvd.	Monona	608555387	Lucky
Jean Coleman	105 N. Lake St.	Monona	6085552654	Max Samantha
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085511749	Basil
Harold Davis	563 Friendly St.	Windsor	6085531794	Iggy
Maria Escobito	345 Maple St.	Madison	608557683	Mulga
Carlos Esteban	2335 Independence La.	Waunakee	608555487	Lucky Sly
George Franklin	110 W. Liberty St.	Madison	608551023	Leo
Peter McTavish	2387 S. Fair Way	Madison	608552765	George
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	Jewel Rosy
David Schroeder	2749 Blackhawk Trail	Madison	6085559435	Freddy

Tag last year

Tag	last year
@wrapper	gradle on top-level to...
	more space in right c...
workspace	Root-Feld muss ping ...

Activity 6 stars 4 watching 3 forks

WEBANWENDUNGEN

Die Webanwendung gibt es nicht

...aber: (fast) alle brauchen JavaScript

...die Frage ist nicht: ob, sondern wo und wieviel



WEBANWENDUNGEN

Die Webanwendung gibt es nicht

...aber: (fast) alle brauchen JavaScript

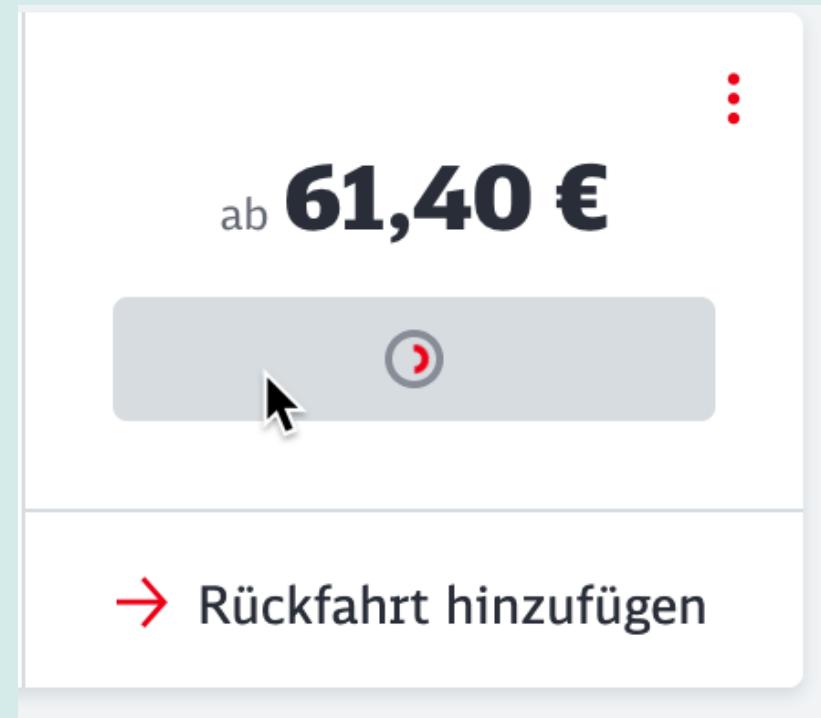
...die Frage ist nicht: ob, sondern wo und wieviel

(...und wer es schreibt oder erzeugt 😈)

Beispiele

Ein Beispiel...

- Ladearimation
- Double-Submit verhindern



Ein Beispiel...

- Feedback direkt beim Tippen
- z.B. Zeichen-Zähler

Your opinion?

Your name:

Your rating:

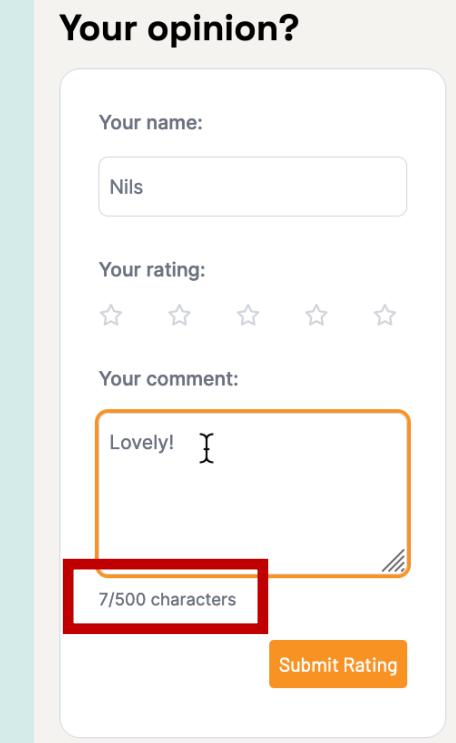
☆ ☆ ☆ ☆ ☆

Your comment:

Lovely!

7/500 characters

Submit Rating



Ein Beispiel...

The screenshot shows a user profile interface with a blue header bar. Below it is a white box containing a user's profile picture and a "Beitrag beginnen" button. At the top right are three buttons: "Medieninhalte" (with a camera icon), "Event" (with a calendar icon), and "Artikel schreiben" (with a text icon). A red box highlights a dropdown menu titled "Feed-Ansicht auswählen: Neueste zuerst" with a downward arrow. A red arrow points to this dropdown from the right. The menu contains two options: "Relevanteste zuerst" and "Neueste zuerst". Below the menu, a note states: "Diese Auswahl wirkt sich nur auf Ihre aktuelle Feed-Ansicht auf diesem Gerät aus. Sie können Ihre Standardansicht in den Einstellungen ändern." At the bottom right of the menu is a "... mehr anzeigen" button.

Beitrag beginnen

Medieninhalte Event Artikel schreiben

Feed-Ansicht auswählen: **Neueste zuerst** ↴

Relevanteste zuerst

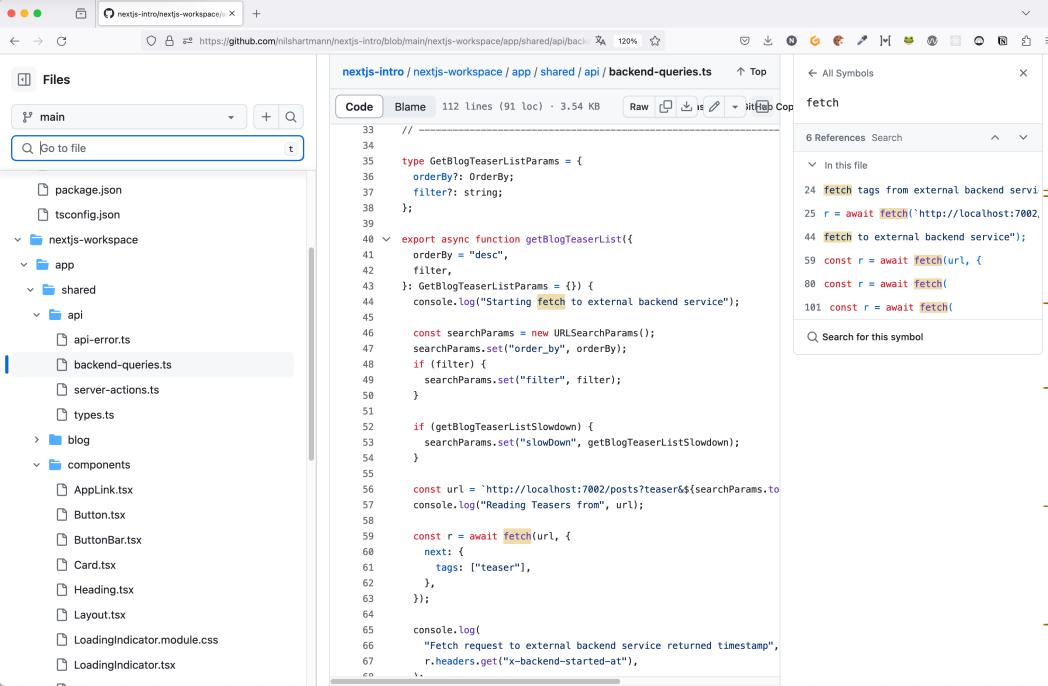
Neueste zuerst

Diese Auswahl wirkt sich nur auf Ihre aktuelle Feed-Ansicht auf diesem Gerät aus. Sie können Ihre Standardansicht in den **Einstellungen** ändern.

... mehr anzeigen

INTERAKTIONEN

Ein Beispiel...



The screenshot shows a browser-based code editor interface. On the left is a sidebar labeled "Files" containing a tree view of files and folders. The "backend-queries.ts" file is selected. The main area displays the code for this file.

```
// -----
type GetBlogTeaserListParams = {
  orderBy?: OrderBy;
  filter?: string;
};

export async function getBlogTeaserList({
  orderBy = "desc",
  filter,
}: GetBlogTeaserListParams = {}): Promise {
  console.log("Starting fetch to external backend service");

  const searchParams = new URLSearchParams();
  searchParams.set("order_by", orderBy);
  if (filter) {
    searchParams.set("filter", filter);
  }

  if (getBlogTeaserListSlowdown) {
    searchParams.set("slowdown", getBlogTeaserListSlowdown);
  }

  const url = `http://localhost:7002/posts?teaser=${searchParams.toString()}`;
  console.log("Reading Teasers from", url);

  const r = await fetch(url, {
    next: {
      tags: ["teaser"],
    },
  });

  console.log(
    "Fetch request to external backend service returned timestamp",
    r.headers.get("x-backend-started-at"),
  );
}
```

Interaktionen: oft nur mit JavaScript umsetzbar

- Es gibt Fortschritte in CSS und HTML, die JS an mehr Stellen verzichtbar machen, aber nicht an allen
- JS braucht man vielleicht schneller/früher als man häufig denkt

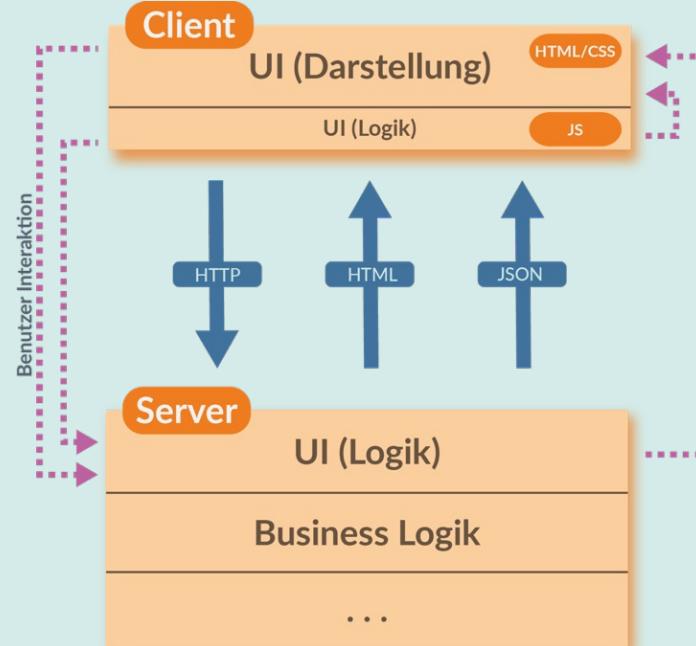
"Klassische" Web-Anwendung (Web 2.0)

Mit JavaScript-Schnipseln (früher jQuery):

- Interaktion
- Validierung

Konsequenz:

- Unsaubere Architektur (was ist wo?)
- Wartungshölle

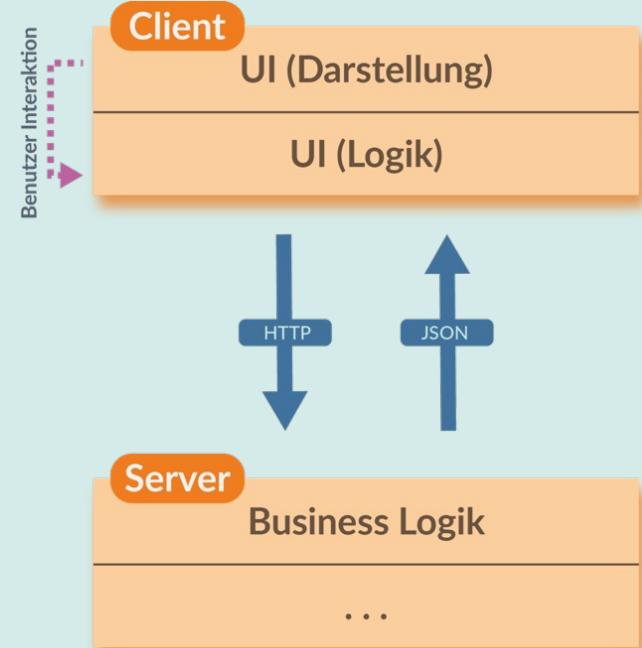


Single-Page-Anwendungen (ca. seit 2010)

- Client ist vollständig in JavaScript geschrieben
- Client läuft vollständig im Browser

Konsequenz:

- Klare Trennung zwischen Backend und Frontend
- Server "nur" für Geschäftslogik, Datenhaltung...
- Frontend "nur" für Darstellung und Interaktion
- Server muss APIs für Daten bereit stellen



Gängige Aussagen über Single-Page-Anwendungen

- JavaScript total schlimm! ("undefined is not a function")
- SEO geht nicht / schlecht!
- Langsam!
- Viel JavaScript muss geladen werden!
- JS-Bibliotheken ändern sich ständig!
- Tooling funktioniert nicht bzw. ist komplex!

HTMX

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/>

Single-Page-Anwendung oder HTMX?

- SPA ist ein **Pattern** oder eine **Architektur**, aber kein konkretes Framework
- Wir müssten bei einem Vergleich eigentlich ein SPA-Framework wählen
 - d.h. es müsste heißen "Angular vs. HTMX" oder "React vs. HTMX"
- Ich versuche eine Mischung aus "Konzept" / "Idee" und Umsetzung (React)

Single-Page-Anwendung oder HTMX?

- HTMX ist **clientseitige** Bibliothek
- Bei einem realistischen Vergleich muss man auch die **Serverseite** betrachten
- Hier haben wir ein ähnliches Problem wie bei SPA
 - es müsste heißen "React vs. HTMX mit Spring und Thymeleaf"
- Auch hier versuche ich einen Mittelweg

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
- Der Server muss HTML-Schnipsel liefern ("Hypermedia")

```
<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

Demo: Hello World

- Das Ergebnis:
Reponse: localhost:8080/hello

HelloWorldController

- HTMX-Attribute:
Request: hx-trigger und hx-get

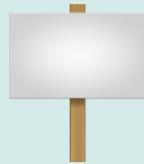
- Der Server muss HTML-Schnipsel liefern
("Hypermedia")

```
<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>
```

Demo



[HTTP://LOCALHOST:8090](http://localhost:8090)

Beispiel-Anwendung

Source-Code (GitHub):

<https://react.schule/bedcon-htmxA>

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

Beispiel 1: Daten nachladen

- Daten erst auf dem Server laden
 - (in HTMX: UI)
- Lade-Animation
- Paginierung

Feedback

Bryson Turner 20. September 2023

Baked garlic butter shrimp is a quick and flavorful dish. Perfect for seafood lovers!

Ivy Turner 10. Juli 2023

Family favorite! Everyone asked for seconds. This recipe is now in our regular rotation.



Beispiel 2: Einfaches Formular

- Wie werden die Daten gesendet?
- Was kommt als Antwort?
- Wie disablen wir das Formular?
- Wie setzen wir eine (Fehler-)Meldung zurück

Subscribe to newsletter

E-Mail

Subscribe

Beispiel 3: Suchfeld

- Wie starten wir einen Server-Request automatisch?
- Wie erweitern wir eine bestehende Liste?
- Wie können wir die Suche zurücksetzen?
- Wie können wir eine Information ein- und ausblenden?
- Mehrere Stellen in einer Seite aktualisieren

Clear

Recipes for sal

LUNCH

Baked Salmon



▼

BREAKFAST

Caesar Salad



▼

Find more...

Beispiel 4: 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

Fazit

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

Single-Page-Anwendung oder HTMX

- It depends (natürlich 😐)

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

Single-Page-Anwendung oder HTMX

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

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

Single-Page-Anwendung 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

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)

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

HTMX

- HTMX hat eine eigene (dogmatische) Sichtweise auf die Welt
- Ob die zu einer Anwendung passt, hängt von der Anwendung ab
- Design und API erinnern an jQuery
- Wollen wir Anwendungen von heute mit Technik von vor zwanzig Jahren bauen?

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

HTMX

- Über das Backend und den Entwicklungsprozess sagt HTMX (fast) nichts
 - Wie sieht es mit Testen, Debuggen, Typsicherheit aus?
 - Habt ihr ein Backend, das HTML-Schnipsel liefern kann?
 - Habt ihr eine passende Template-Sprache?
- **Für Spring Boot gibt es mittlerweile built-in Support**

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

HTMX

- Für Kleinigkeiten muss ein Server Request für neue UI gemacht werden
 - Ist das ein Entwicklungs- und Architekturmuster das dir gefällt?
 - Wie sieht es mit Latenz und Datenmenge aus?
 - Wie viele Stellen hast du, an denen das nicht funktioniert?
 - Zum Beispiel Eingabefeld leermachen, Zeichenzähler...
- Wie gehst du mit Stellen um, an denen du doch JavaScript braucht?

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

Single-Page-Anwendung

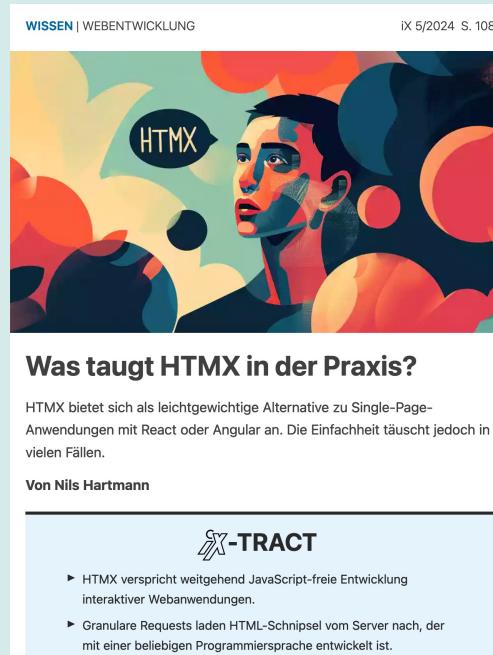
- Adressieren das Problem "echter" Anwendungen im Browser
- Verwenden die Sprache der Ziel-Platform

FAZIT: SINGLE-PAGE-ANWENDUNG ODER HTMX?

Single-Page-Anwendung

- Ausgereifte Technologie
 - ...mit allen, was wir aus dem Backend kennen (automatische Tests, Typsicherheit, Debugger, CI-Pipelines)
- Stand der Technik nicht mit 2010 vergleichbar
 - Vieles hat sich konsolidiert und stabilisiert
 - Trotzdem noch viel Wandel und Innovation
- Als Alternative erscheinen mir Fullstack-Frameworks konsequenter als HTMX

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



Vielen Dank!

Code: <https://react.schule/bedcon-htmrx>

Slides: <https://react.schule/bedcon-htmrx-slides>

heise Artikel: <https://react.schule/heise-htmrx>

Fragen & Kontakt: nils@nilshartmann.net

Twitter: [@nilshartmann](https://twitter.com/nilshartmann)