

DATA-DRIVEN UX

Core Web Vitals und das Rail performance Model

Moin 🖐️

Ich bin Jakob,
und manchmal mache ich 1 WebDev.

- Hab ne Website: runjak.codes
- Gibt Slides: github.com/runjak/nook2022

Es gibt da dann ja so Komplexitäten, mit Software

- im Problem das wir lösen wollen
- in Browsern
- in der UI

Komplexität ist immer mal sehr verlockend.
(Wir denken uns dann was)

Angenehm kann dann auch sein

- eine gute Lösung zu bauen
- UI, die schwuppt
- Software, die Leute gerne benutzen

UX

The user experience (UX) is how a user interacts with and experiences a product, system or service ...

en.wikipedia.org/wiki/User_experience

a user's perceptions and responses that result from the use and/or anticipated use ...

ISO 9241

Ganz schöner Themenkomplex.

Wir gucken uns nur ein paar Aspekte an,
Google tut da was:

- Das Rail performance Modell
- Core Web Vitals

Wir gucken uns nicht an:

- A11y, aria
- reduced-motion
- high-contrast / low-contrast
- Viele Sachen

HOT TAKE 🌶️

- Gute UX passiert nicht einfach so
- Es reicht nicht aus, auf Gutes zu hoffen
- Gute Software hat Methode
- Messen und Verbessern von Mängeln
- Engineering culture

RESPONSE TIMES

(Die für Menschen)

Response time in man-computer conversational transactions

by **ROBERT B. MILLER**

International Business Machines Corporation
Poughkeepsie, New York

Fall Joint Computer Conference, 1968

VIEL INHALT

- Activity clumping
- Psychological closure
- Stepdown discontinuities bei Verzögerungen

BEISPIELE

- Zeit bis zum Signalton am Telefon
- Nebenläufige Prozesse - in 1h oder 1t ist ~egal
-  +  -> 

BEHAUPTUNG

Reaktionszeiten verändern sich wenig

-> Empfehlungen

NIELSEN NORMAN GROUP

Jakob Nielsen: Usability Engineering, 1993

www.nngroup.com/../response-times-3-important-limits/

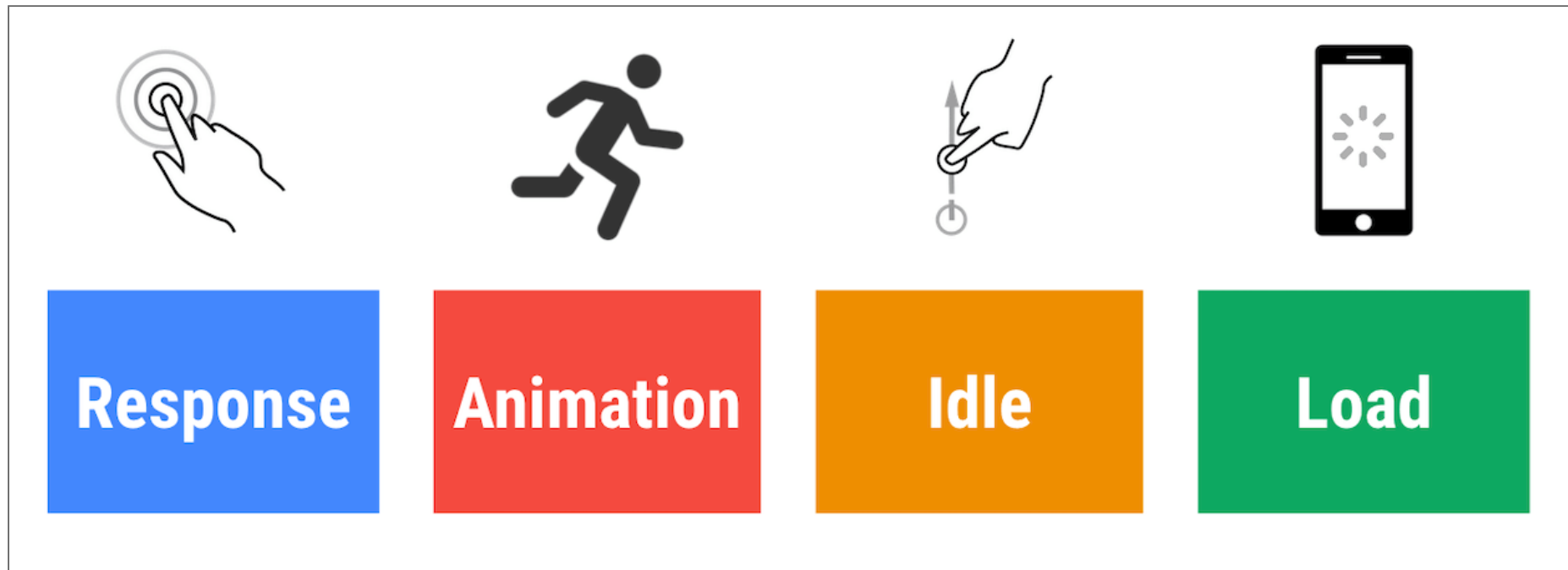
OBERE GRENZEN

- 0.1s um eine Reaktion als Sofort wahrzunehmen
- 1s für einen ununterbrochenen Gedankenfluss
- 10s für die Aufmerksamkeit

Nach wie vor Grundlage für viele Interfaces

Davon wurde auch ein spezifisches Performancebudget abgeleitet, das RAIL Modell.

DAS RAIL MODELL



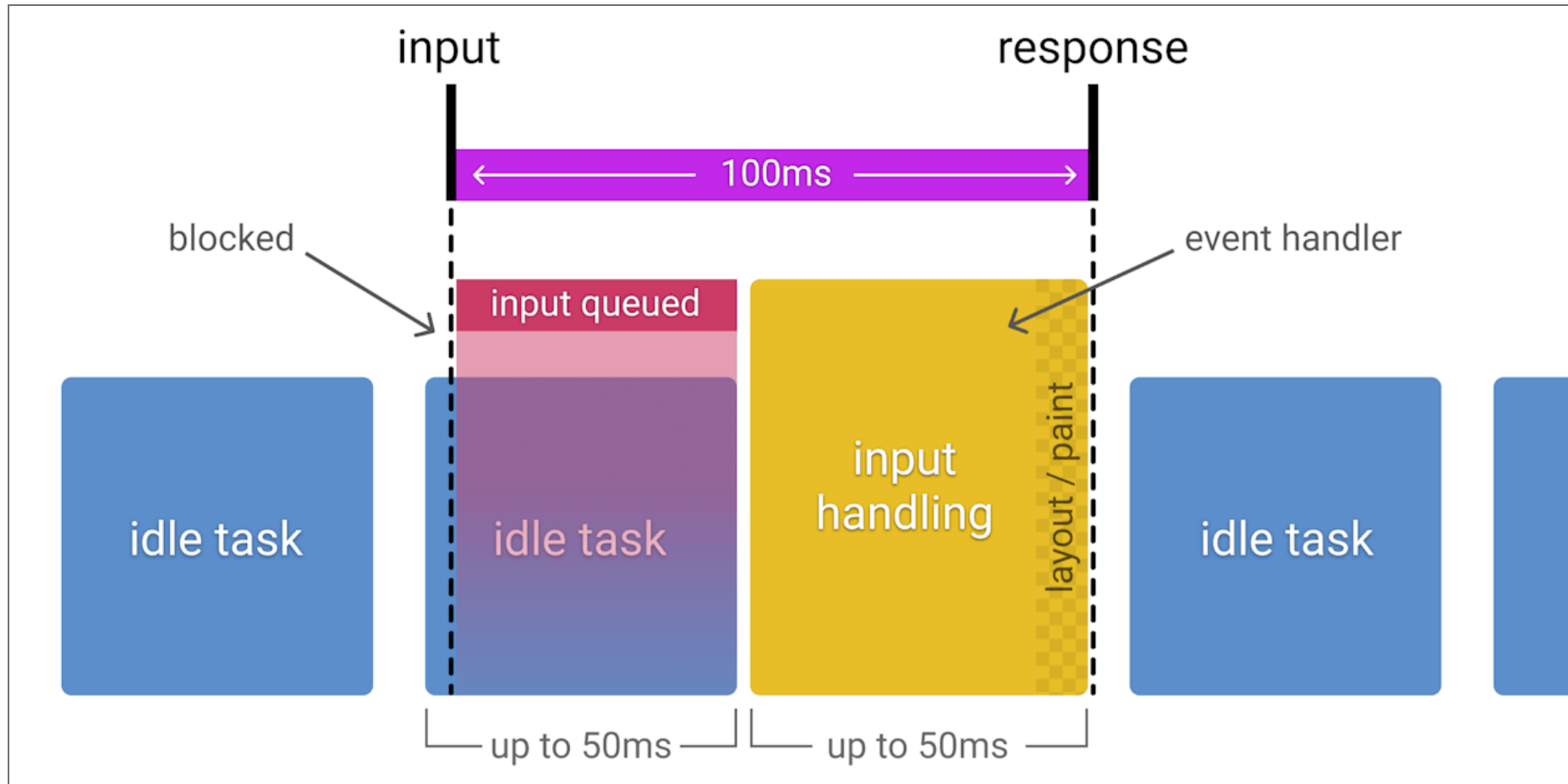
web.dev/rail

IDLE

- : Idlezeit maximieren
- In der Idlezeit angestaute Arbeit erledigen
- Arbeit auf 50ms Blöcke beschränken

RESPONSE

- : 0.1s Budget
 - 100ms
- Klicks, Tastendrücke und sowas



ANIMATION

- : 0.1s Budget
- Zeit für JS pro Frame: $100\text{ms}/60 \rightarrow 16\text{ms}$
 - 6ms für Rendering
 - $\leq 10\text{ ms}$ für JS

LOAD

- : 1s, 10s Limits
- ≤ 5 s zum initialen Laden der Seite
- ≤ 2 s für weitere loads

FOKUS AUF USER

- Daten von Nutzenden in Betracht ziehen.
- Tunnel in Brandenburg?

[wikipedia../File:Telefon BW 2012-02-18 13-44-32.JPG](#)

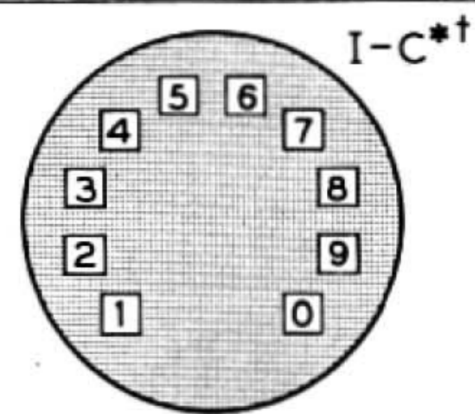
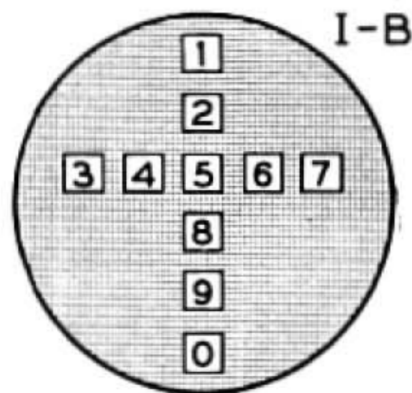
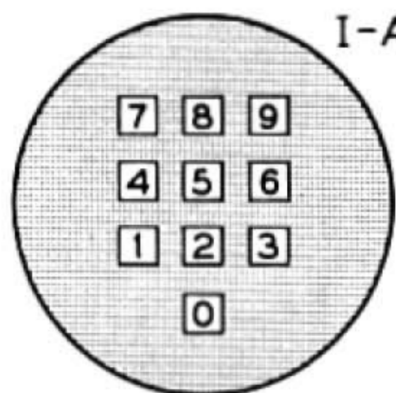


Human Factors Engineering Studies of the Design and Use of Pushbutton Telephone Sets

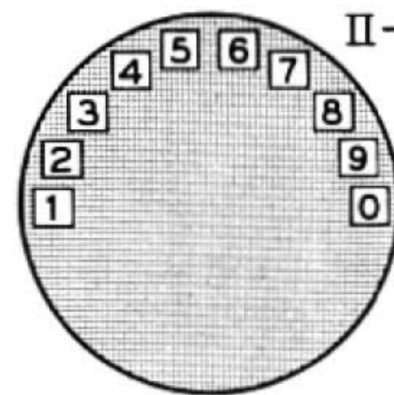
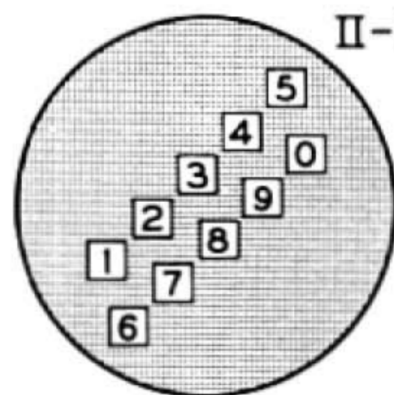
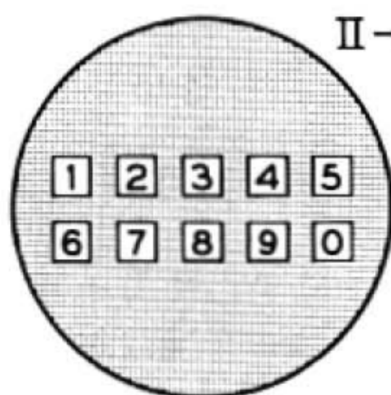
By R. L. DEININGER

(Manuscript received February 16, 1960)

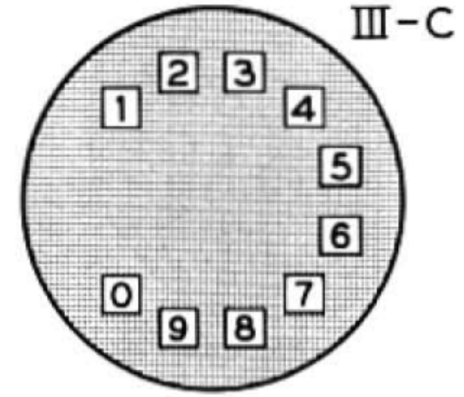
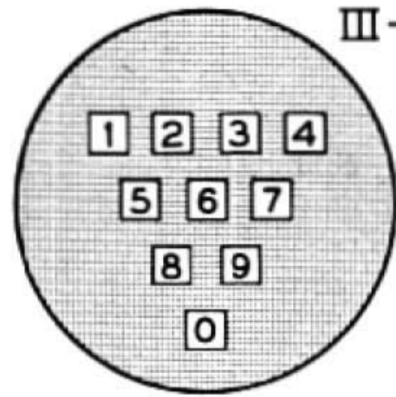
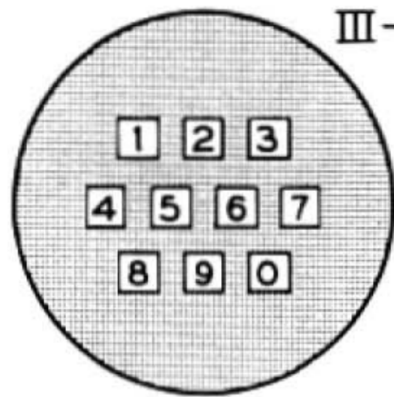
GROUP I



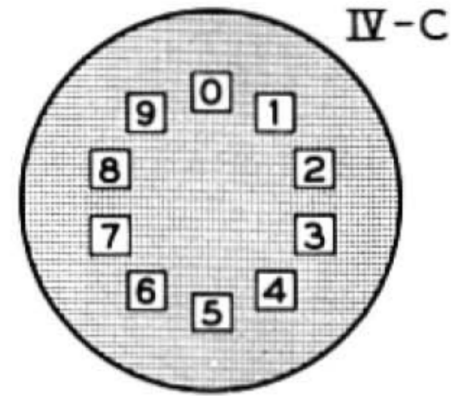
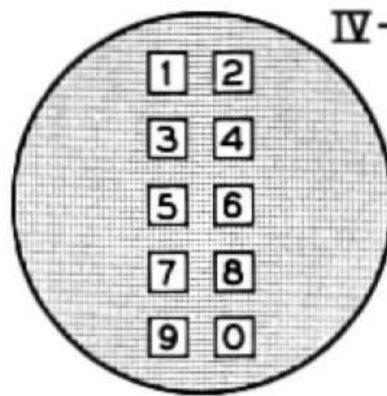
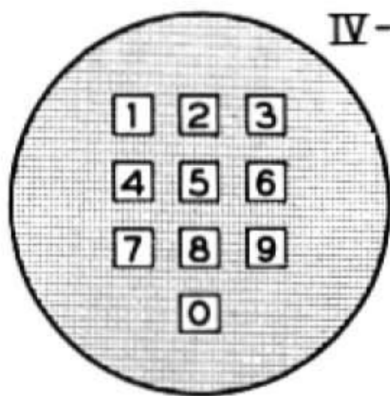
GROUP II



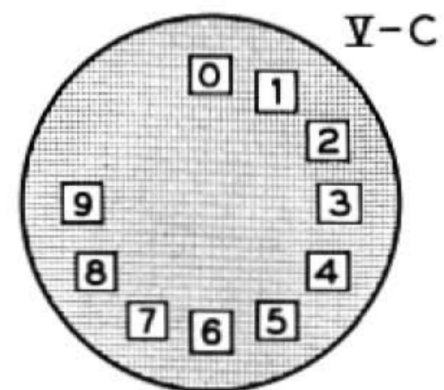
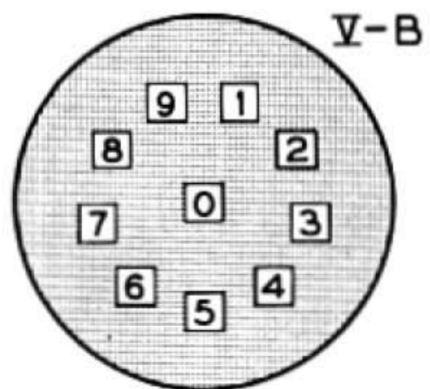
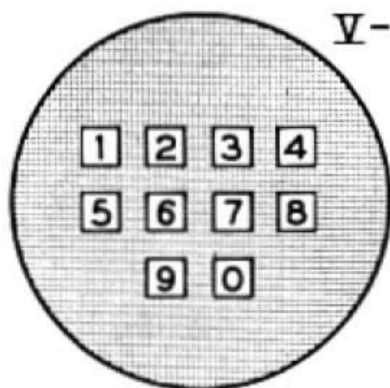
GROUP III



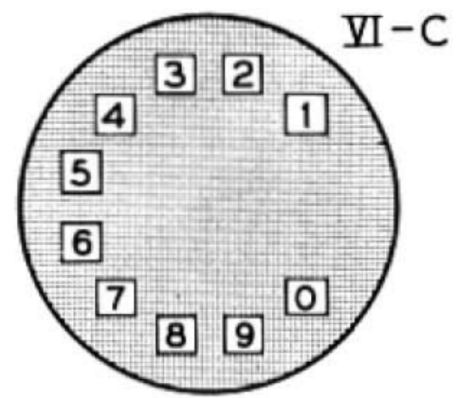
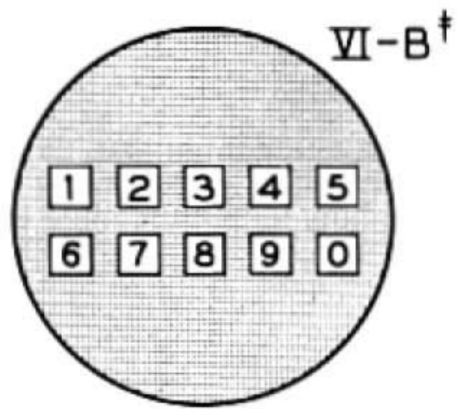
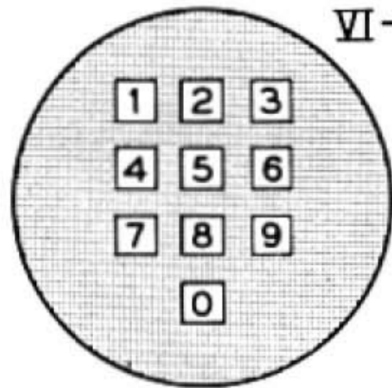
GROUP IV

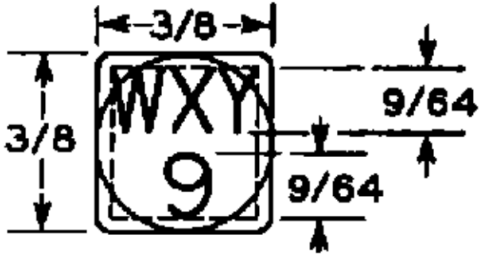
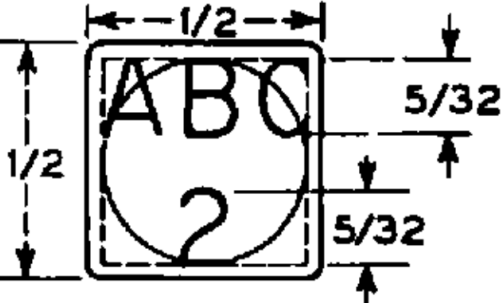


GROUP V



GROUP VI



BUTTON ARRANGEMENT	SET AND SPACING	BUTTON AND LETTERING SIZE	REMARKS
 (3X3+1)	SET NO. 1: 3/4 IN. BETWEEN BUTTON CENTERS		KEYING TIME: 6.35 SEC KEYING ERRORS: 7.1% 2 VOTES FOR 9 VOTES AGAINST
	SET NO. 2: 3/4 IN. BETWEEN BUTTON CENTERS		KEYING TIME: 5.83 SEC KEYING ERRORS: 1.3% 1 VOTE FOR 1 VOTE AGAINST

Datenerhebung:

- Korrektheit der Eingaben
- Eingabegeschwindigkeiten
- Ratschläge/Ansichten der Testenden in Textform

WEB VITALS

web.dev/vitals

- Lab data
- Field data
- p75

WER ERHOFFT SICH HIER WAS?

- User
- WebDevs
- Google

LCP



web.dev/lcp

LARGEST CONTENTFUL PAIT (LCP)

- Größtes Element -> wichtigster Inhalt
- Bild, Video, Text, ..

FID



web.dev/fid

FIRST INPUT DELAY (FID)

- Verzögerungen stören
- JavaScript  
- Langsame Verbindungen / viele Daten?

CLS



web.dev/cls

CUMULATIVE LAYOUT SHIFT (CLS)

- Verschiebungen ohne Interaktion
- Verschiebungsscores
- Summe über einen Zeitraum

KURZES RANTEN

- Karusselle
- Notifications, die reingeschoben werden
- Automatisches Erneuern von Inhalten
- Bilder, Fonts
- ...

Es gibt da ne Sammlung von guten Patterns
web.dev/patterns/web-vitals-patterns

MESSEN

So, hier - wie messen wir das nun?

- **Chrome UX Report**
- Begrenzt mit Lighthouse
- Selbst nachgucken (DevTools)
- Selbst sammeln

SELBST MESSUNGEN SAMMELN

- npmjs.com/package/web-vitals
- [Navigator.sendBeacon](#)
- Aggregation im Backend

DATENSPAARSAMKEIT

- Messwert
- Seitentyp, Route oder sowas
- Deviceklasse

- Lasst mal messen 🕒
- Lasst mal verbessern 🚀
Weil wir Software mögen und so 😊