Engineer

Webbrowser

programmet, man bruger til at åbne websider

Hvad det er

En webbrowser er det program, der slår en adresse op, henter det, serveren svarer med, og bygger en side ud af det, som man kan læse, rulle i og klikke på. Til daglig hedder den bare browseren. Chrome, Safari, Firefox og Edge er navnene på ikonerne, men under dem ligger langt færre programmer, end navnene lader ane.

HTML-standarden bruger sjældent ordet browser. Den taler om en *user agent*, altså den software, der handler på læserens vegne, og det meste af standarden er en opremsning af, hvad sådan en skal gøre med en side. Tre af pligterne forklarer det meste af, hvad en browser er til.

Den første er at fortolke koden. Standarden beskriver, hvordan et HTML-dokument skal læses, uanset om det er skrevet korrekt eller ej. Et manglende sluttag eller et element på det forkerte sted standser derfor ikke siden, og alle browsere, der følger standarden, reparerer den samme fejlbehæftede kode til det samme træ af elementer. Derfor vises en sjusket side alligevel, og derfor kan fejlene i den leve i årevis, uden at nogen bemærker dem.

Den anden er hændelsesløkken, på engelsk *event loop*. Klik, tastetryk, timere, svar fra netværket, scripts og tegningen af næste billede stiller sig i kø og skiftes til at komme til. Et script, der holder løkken i to sekunder, holder også klik, indtastning og næste opdatering af skærmen tilbage, og det er præcis det, folk mener, når de siger, at en side er gået i stå.

Den tredje er at holde websteder adskilt. Standarden definerer en oprindelse, en *origin*, som et skema, en vært og en port, og et site som skema plus den registrerbare del af domænet. Et script fra én oprindelse, der rækker ind i

dokumentet fra en anden, får en sikkerhedsfejl i stedet for indholdet. Det er i høj grad den regel, håndhævet af browseren ved hver eneste adgang, der holder netbanken i én fane fri af avisen i den næste.

Selve arbejdet udføres af browserens *motor*. De fleste browsere i brug i dag er bygget på én af tre: Blink i Chrome, Edge og en lang række andre, WebKit i Safari og Gecko i Firefox. Et nyt browsernavn betyder som regel en ny skal om en af dem, med sin egen adresselinje, sine egne indstillinger og sin egen synkronisering, men de samme regler for at læse og tegne siden.

To naboer bliver ofte forvekslet med den. En søgemaskine er et website, som browseren åbner, når adresselinjen får ord i stedet for en adresse. En *webview* er en browsermotor bygget ind i en anden app, uden adresselinje og uden de fleste knapper. Testværktøjer tilføjer en browser uden synlig brugergrænseflade, en *headless* browser, som tegner sider, ingen kigger på, så et script kan tjekke dem.

Brug begrebet i en beslutning

Brug »Hvad er en webbrowser?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Brugerflader, der indlæses hurtigt, virker på en telefon og opfører sig ens i alle de browsere, jeres kunder nu bruger.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har styret full-stack-udviklingen af GIS-kort — full stack-kortløsning, fra database til browser

Har styret full-stack-udviklingen af GIS-kort Kontakt os →

Hvorfor det betyder noget

Serveren kan sende præcis de rigtige bytes, og læseren kan alligevel få en ulæselig side. Det, man ser, er nemlig det, ens browser har gjort ud af de bytes, med ens egne indstillinger lagt ovenpå. Det er i browseren, arbejdet bliver bedømt.

Man afprøver motorer, ikke mærkenavne. En testplan med fire browsernavne kan dække to motorer to gange og overse den tredje. En brugbar liste navngiver motorerne og de steder, hvor den samme motor opfører sig anderledes, for eksempel på en telefon med touchskærm eller i en printdialog.

Læserens indstillinger er en del af siden. Browseren bærer et foretrukket farveskema, et ønske om mindre bevægelse, et zoomniveau, en standardskriftstørrelse og nogle gange en indbygget oversættelse. En side, der ser bort fra det hele, er lavet til én opsætning hos én person, som regel den, der byggede den.

Man spørger efter funktionen i stedet for at gætte. Motorerne får ny CSS og nye grænseflader på forskellige tidspunkter. En side, der har brug for noget, som kun nogle motorer kan, kan spørge motoren direkte: CSS har `@supports`, og et script kan teste for en egenskab, før det bruger den. User-agent-strengen er en dårlig erstatning, for browsere har længe skrevet den, så de lignede hinanden og slap forbi ældre kode, der kiggede efter navne.

Regler i browseren er en høflighed. Formularvalidering, en deaktiveret knap og et skjult link kører alle i et program, som læseren ejer og kan ændre. De gør den ærlige vej lettere, men de standser ingen, der åbner udviklerværktøjerne eller springer siden helt over. Alt, hvad der beskytter penge, data eller adgang, tjekkes derfor igen på serveren.

Sådan ser det ud i praksis

Dette website understreger sine links med skip-ink, indstillingen, der bryder linjen, hvor den ville krydse en underlænge. I Chromium så bruddene rigtige ud. Målt i en rigtig Safari ved otte gange forstørrelse var de omkring seks gange smallere, og årsagen var en fejl i WebKit: ét enkelt skriftsnit, der erklærede et ikke-latinsk tegninterval, forringede skip-ink for hele familien. Løsningen blev et omdøbt familienavn i fontstakken. Beretningen om Safaris understregning og skriftudsnittet bag den slutter med den pointe, denne artikel ville sætte først: en skip-ink-måling gælder kun for den motor, der frembragte den. Tabellen, diagnosen og den første rettelse var alle lavet i Chromium, og skærmbilledet, der genåbnede sagen, kom fra et menneske, der kiggede på siden.

Udskrift er et andet sted, hvor motorerne går hver sin vej. Det print-stylesheet på 637 linjer beskriver fire former for motoradfærd som fælder, hver med en reel defekt bag sig. Viewport-enheder regnes ud fra sideboksen i én motor og fra vinduet i en anden, så en side printet fra et bredt vindue mistede en tredjedel af hver linje. Fastplacerede elementer tegnes på hvert eneste ark. Etiketter, der var farvet til det mørke standardtema, kom ud hvidt på hvidt hos læsere, hvis browser foretrak mørkt. Og en flex- eller grid-boks, som paginerede medier ikke kan dele, gav et blankt første ark. Tjekket bag det prøver seks kombinationer af

papir og orientering under begge farveskemaer og tæller de faktiske sider, og siden noterer stadig én motor, der ignorerer kontrollen af enlige linjer øverst og nederst på siden.

Det tredje tilfælde fandt læserne selv. Da sitet blev lagt ud som et Tor onion-spejl på en læsbar adresse, sad de første læsere på en anden browsermotor, som ikke implementerer den scroll-drevne animation, footeren byggede på, og hver eneste af dem fik kolofonen malet hen over forsiden. Defekten fandtes også på det almindelige site. Der skulle bare et publikum på en anden motor til at vise den.

Hvor det går galt

- **En side afprøves i browseren på udviklerens skrivebord og erklæres færdig.** Hver vane hos den ene motor bliver til en skjult antagelse. Modtrækket er en kort matrix med én browser pr. motor plus en telefon, og en rettelse er først færdig, når den er målt i dem alle.
- **Koden læser browserens navn og gætter på, hvad den kan.** Gættet rammer forkert for hver browser, der låner en andens etiket, og det forældes med hver ny version. Spørg efter funktionen i stedet. Reglen uden for blokken er den, alle andre motorer får, så den skal være den sikre tilstand:

```
.footer {
    position: static;
}

@supports (animation-timeline: scroll()) {
    .footer {
        animation: settle linear both;
        animation-timeline: scroll();
    }
}
```

- **Elementer uden stil arver browserens egne standarder.** Overskrifter, marginer og især formularfelter kommer med hver motors indbyggede stylesheet, og de er forskellige. Et design, der så færdigt ud, kan hvile på én motors valg.
- **En regel, der kun håndhæves på siden, regnes for håndhævet.** En knap, som et script har gjort grå, et felt, der kun tager tal, et link skjult for folk uden abonnement: hver af dem holder kun for læsere, der lader siden være, som den blev sendt. Serveren gentager hvert tjek, der betyder noget.

- **Links åbner inde i en andens app.** Et link, man trykker på i en chat eller et socialt feed, lander ofte i en indbygget browser eller en webview, tit med sit eget lager, så en person, der er logget ind i sin almindelige browser, ankommer som fremmed. Forløb, der skifter til en anden app eller bygger på en eksisterende session, skal også afprøves derfra.

Læs mere

- NIST CSRC Glossary: web browser — to korte definitioner fra NIST's egne publikationer, den ene blot »client software used to view Web content«.
- HTML Standard: Parsing HTML documents — reglerne, alle browsere følger for at gøre kode, fejlfri eller ej, til det samme træ.
- HTML Standard: Event loops — hvordan input, scripts, svar fra netværket og tegning af siden skiftes til at køre.
- HTML Standard: Loading web pages — oprindelser, sites og reglerne, der forhindrer én side i at læse en anden.
- WebKit Bugzilla: bug 255159 — motorens egen registrering af den skip-ink-fejl, der er beskrevet ovenfor.

Ordliste

Vælg ud fra jeres faktiske vilkår

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er en webbrowser?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/browser/