Engineer

Nettleser

programmet en person bruker for å åpne nettsider

Hva det er

En nettleser er programmet som tar imot en adresse, henter det serveren sender tilbake, og gjør det om til en side du kan lese, rulle i og klikke på. Mange norske utviklere sier bare browser, etter engelsk *web browser*. Chrome, Safari, Firefox og Edge er navnene på ikonene. Under dem ligger langt færre programmer enn navnene gir inntrykk av.

HTML-standarden bruker sjelden ordet nettleser. Der heter det *user agent*, programvaren som handler på leserens vegne, og mesteparten av standarden er en liste over hva en slik agent skal gjøre med en side. Tre av pliktene forklarer det meste av hva en nettleser er til for.

Den tolker koden. Standarden gir regler for hvordan HTML-dokumenter skal leses «whether they are syntactically correct or not», altså uansett om de er skrevet riktig. Et manglende sluttagg eller et element på feil sted stopper derfor ikke siden. Alle nettlesere som følger standarden, reparerer den samme ødelagte koden til det samme treet av elementer. Derfor vises en slurvete side likevel, og derfor kan feilene i den bli stående i årevis uten at noen legger merke til dem.

Den kjører en hendelsesløkke, på engelsk *event loop*. Klikk, tastetrykk, tidtakere, svar fra nettverket, skript og tegningen av neste bilde stiller seg i kø og venter på tur. Et skript som holder løkken i to sekunder, holder også igjen klikk, skriving og neste oppdatering av skjermen. Det er akkurat det folk mener når de sier at en side har hengt seg.

Den holder nettsteder fra hverandre. Standarden definerer et opphav, en *origin*, som et skjema (som `https`), en vert og en port, og et *site* som skjemaet pluss den registrerbare delen av domenet. Et skript fra ett opphav som griper inn i

dokumentet til et annet, får en sikkerhetsfeil i stedet for innholdet. Mye av det som holder nettbanken i én fane trygg for nyhetssiden i den neste, er nettopp denne regelen, håndhevet av nettleseren ved hver eneste tilgang.

Delen som gjør arbeidet, er *motoren*. De fleste nettlesere i bruk i dag er bygget på én av tre: Blink i Chrome, Edge og en lang hale av andre, WebKit i Safari og Gecko i Firefox. Et nytt nettlesernavn betyr som regel et nytt skall rundt en av disse, med egen adresselinje, egne innstillinger og egen synkronisering, men med de samme reglene for å lese og tegne en side.

To naboer blir ofte forvekslet med den. En søkemotor er et nettsted som nettleseren åpner når adresselinjen får ord i stedet for en adresse. En *webview* er en nettlesermotor bygget inn i en annen app, med adresselinjen og de fleste kontrollene fjernet. Testverktøy legger til en hodeløs nettleser, på engelsk *headless*, som tegner sider ingen ser på, slik at et skript kan sjekke dem.

Bruk begrepet i en beslutning

Bruk «Hva er en nettleser?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Grensesnitt som laster raskt, fungerer på mobilen og oppfører seg likt i alle nettlesere kundene dine måtte bruke.

Se tjenestene våre Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Styrte fullstack-utviklingen av GIS-kart — fullstack-kartutvikling, fra database til nettleser

Styrte fullstack-utviklingen av GIS-kart Ta kontakt →

Hvorfor det er viktig

Serveren kan sende nøyaktig de riktige bytene, og leseren kan likevel få en ulesbar side. Det en person ser, er det nettleseren gjorde ut av bytene, med personens egne innstillinger lagt oppå. Det er i nettleseren arbeidet blir bedømt.

Test motorer, ikke merkenavn. En testplan med fire nettlesernavn kan dekke to motorer to ganger og gå glipp av den tredje. En nyttig liste navngir motorene og plattformene der samme motor oppfører seg annerledes, f.eks. en telefon med liten skjerm og berøring, eller en utskriftsdialog.

Leserens innstillinger er en del av siden. En nettleser bærer med seg et foretrukket fargevalg, et ønske om mindre bevegelse, et zoomnivå, en standard tekststørrelse og noen ganger et oversettelseslag. En side som overser alt dette,

er laget for én konfigurasjon hos én person, som regel den som bygde siden.

Spør etter funksjonen i stedet for å anta den. Motorene får ny CSS og nye grensesnitt til ulike tider. En side som trenger noe bare noen motorer kan, kan spørre motoren direkte: CSS har `@supports`, og et skript kan teste om en egenskap finnes før det bruker den. User-agent-strengen er en dårlig erstatning. Nettlesere har lenge skrevet den delvis for å ligne hverandre, slik at eldre kode som snuste etter navn, skulle slippe dem inn.

Regler i nettleseren er høflighet. Validering av et felt, en deaktivert kontroll eller en skjult lenke kjører alle i et program som leseren eier og kan endre. De gjør den ærlige veien enklere, men stopper ingen som åpner utviklerverktøyene eller hopper over siden helt. Alt som beskytter penger, data eller tilgang, sjekkes derfor på nytt på serveren.

Slik ser det ut i praksis

Dette nettstedet understreker lenkene sine med skip-ink, innstillingen som bryter linjen der den ville krysset en underlengde. I Chromium så bruddene riktige ut. Målt i en ekte Safari ved åttedobbel skala var de rundt seks ganger smalere, og årsaken viste seg å være en feil i WebKit: ett enkelt snitt som oppga et ikke-latinsk tegnområde, svekket skip-ink for hele familien. Løsningen ble et nytt familienavn i fontstakken. Beretningen om understrekingen i Safari og fontdelsettet bak den slutter med lærdommen denne artikkelen setter først: en skip-ink-måling gjelder bare for motoren som laget den. Tabellen, diagnosen og den første rettelsen var alle Chromium, og skjermbildet som gjenåpnet saken, kom fra et menneske som så på siden.

Utskrift er et annet sted der motorene går hver sin vei. Utskriftsstilarket på 637 linjer dokumenterer fire motoroppførsler som feller, hver knyttet til en reell feil. Motorene måler visningsområdeenheter på papir mot hver sin ting, én mot sideboksen og en annen mot vinduet, og hos den siste mistet en side skrevet ut fra et bredt vindu en tredjedel av hver linje. Elementer med fast posisjon males på hvert ark. Etiketter farget for det mørke standardtemaet kom ut hvitt på hvitt hos lesere med nettleseren innstilt på mørkt. En flex- eller grid-boks som sidedelte medier ikke kan dele, ga et tomt første ark. Kontrollen som vokter dette, kjører seks kombinasjoner av papir og retning under begge fargevalgsinnstillingene og teller de faktiske sidene, og siden noterer fortsatt én motor som ignorerer kontroll av horunger og enker.

Det tredje tilfellet fant leserne selv, før noen test gjorde det. Da nettstedet ble lagt ut som et onion-speil for Tor på en lesbar adresse, brukte de første leserne en annen nettlesermotor, en som ikke implementerer den rullestyrte animasjonen bunnteksten var avhengig av. Hver eneste av dem fikk kolofonen malt over hele forsiden. Det vanlige nettstedet hadde den samme feilen; det trengtes et publikum på en annen motor for å vise den.

Hvor det går galt

- **En side sjekkes i nettleseren på utviklerens skrivebord og erklæres ferdig.** Hver vane i den ene motoren blir en skjult antakelse. Botemiddelet er en kort matrise med én nettleser per motor, pluss en telefon, og en rettelse er først ferdig når den er målt i hver av dem.
- **Koden leser nettleserens navn og gjetter hva den kan.** Gjetningen slår feil for hver nettleser som låner en annens merkelapp, og den går ut på dato med hver ny versjon. Å spørre etter funksjonen lar motoren selv avgjøre, og regelen utenfor blokken er den alle andre motorer får, så den bør være den trygge tilstanden:

```
.footer {
    position: static;
}

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

- **Elementer uten stil arver nettleserens egne standardverdier.** Overskrifter, marger og særlig skjemafelt kommer med hver motors innebygde stilark, og de er ulike. Et design som så ferdig ut, kan ha lent seg på én motors valg uten at noen har bestemt dem.
- **En regel som bare håndheves på siden, regnes som håndhevet.** En knapp gjort grå av et skript, et felt som bare tar tall, en lenke skjult for folk uten abonnement: hver av dem holder bare for lesere som lar siden være slik den ble sendt. Serveren gjentar hver sjekk som betyr noe.
- **Lenker åpnes inne i andres apper.** En lenke du trykker på i en chat eller et sosialt medium, havner ofte i en innebygd nettleser eller en webview, gjerne med egen lagring. En person som er logget inn i sin vanlige nettleser, kommer da fram som en fremmed. Flyter som hopper over til en annen app eller bygger på en eksisterende økt, må også testes fra disse inngangene.

Les mer

- NIST CSRC Glossary: web browser — to korte definisjoner hentet fra NIST-publikasjoner, den ene den enkle «client software used to view Web content».
- HTML Standard: Parsing HTML documents — reglene alle nettlesere følger for å gjøre kode, ødelagt eller ikke, om til det samme treet.
- HTML Standard: Event loops — hvordan input, skript, nettverkssvar og tegning av siden bytter på å kjøre i én løkke.
- HTML Standard: Loading web pages — opphav, *sites* og reglene som hindrer én side i å lese en annen.
- WebKit Bugzilla: bug 255159 — motorens egen sak om skip-ink-feilen som er beskrevet ovenfor.

Ordliste

Velg ut fra deres faktiske rammer

Beskriv arbeidsflyten, eksisterende grensesnitt og brukerne som trenger endringen. Vi kan definere en testbar leveranse og en overlevering som gjør teamet i stand til å vedlikeholde løsningen.

Se hva det koster Ta kontakt →

Gjør begrepene praktiske

Spør hvordan «Hva er en nettleser?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://engineer.company/nb/glossary/browser/