Engineer

Webbappsmanifest

en JSON-fil som beskriver en webbplats som installerbar app

Vad det är

Ett webbappsmanifest är ett JSON-dokument som beskriver en webbplats som ett program: vad det heter, vilken adress det öppnas på, vilka adresser som hör till det, hur fönstret ska se ut och vilka bilder som representerar det. En sida pekar ut filen med ett `link` -element vars `rel` är `manifest` , och en webbläsare som erbjuder installation läser filen för att avgöra vad den installerade kopian ska bli. W3C-specifikationen som definierar formatet kallar det bara manifestet, och det gör de flesta i vardagligt tal också.

Filen är vanlig JSON. Dess registrerade medietyp är `application/manifest+json` och dess registrerade filändelse är `.webmanifest` , men specifikationen godtar vilken JSON-medietyp som helst, och många webbplatser serverar en `manifest.json` . Varje fält, members i specifikationens språk, är frivilligt. Ett litet manifest för en påhittad webbplats med tidvattentabeller ser ut så här:

```
{
    "name": "Harbour Tide Tables",
    "short_name": "Tides",
    "start_url": "/tides/",
    "display": "standalone",
    "background_color": "#ffffff",
    "icons": [
        {
            "src": "/icons/tides-512.png",
            "sizes": "512x512",
            "type": "image/png"
        }
    ]
}
```

Tre fält avgör vad appen är, till skillnad från hur den ser ut. `start_url` föreslår sidan som den installerade appen öppnas på; webbläsaren kan ignorera den eller låta användaren ändra den. `scope` är den mängd adresser som räknas som en del av appen. Saknas fältet tar specifikationen startadressen och skär bort filnamn, frågesträng och fragment, så exemplet ovan äger allt under `/tides/` . `id` är appens identitet, och saknas det faller identiteten tillbaka på startadressen. Relativa ikonadresser och startadressen tolkas relativt manifestets egen URL och inte mot sidan som länkade till det. Ett relativt `id` tolkas däremot relativt startadressens ursprung.

Resten är presentation. `display` ber om `fullscreen` , `standalone` , `minimal-ui` eller `browser` , och en webbläsare som inte kan tillmötesgå önskemålet går nedåt i den listan i tur och ordning. Varje ikon kan ha ett `purpose` : `any` som standard, `maskable` för en bild som är ritad för att tåla beskärning, eller `monochrome` för en bild vars färger kastas bort så att bara alfakanalen används. `theme_color` färgar fönstret runt appen. En kompletterande anteckning från W3C lägger till `description` , `categories` och `screenshots` , som installationsdialoger och appbutiker kan visa.

Använd begreppet i ett beslut

Använd ”Vad är ett webbappsmanifest?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Gränssnitt som laddar snabbt, fungerar i mobilen och beter sig likadant i alla webbläsare dina kunder råkar använda.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Styrde full stack-utvecklingen av GIS-kartor — kartutveckling i full stack, från databas till webbläsare

Styrde full stack-utvecklingen av GIS-kartor Hör av er →

Varför det spelar roll

Manifestet ger installationsmetadata som webbplatsen själv styr. Specifikationen lämnar beslutet att installera åt webbläsaren: den får erbjuda ett sätt att installera, och varje webbläsare sätter egna villkor, av vilka en del kräver vissa fält. Ett team levererar underlaget och webbläsaren bestämmer vad den gör med det, så den användbara frågan är inte så mycket om ni ska ha ett manifest som vad det lovar.

Identiteten är det löfte som överlever filen. När en webbläsare hämtar manifestet på nytt säger specifikationen att den bör använda ett matchande `id` som en signal om att den nya filen ersätter manifestet för en app som redan

är installerad. Eftersom ett saknat `id` är lika med startadressen ändrar en flyttad startsida tyst identiteten, så webbläsaren bör behandla den nya filen som en annan app än den som finns på läsarens enhet. Att låsa `id` redan i den första versionen kostar en rad och tar bort den risken.

Omfånget, `scope`, avgör var appfönstret tar slut. Navigering inom omfånget stannar i det installerade fönstret. En länk som leder ut ur det öppnas ändå, men specifikationen förväntar sig att webbläsaren visar att läsaren har lämnat appen, så ett omfång som är snävare än webbplatsen får vanliga länkar mellan dess avdelningar att se ut som utgångar.

Det mesta som syns gå fel gäller ikonerna. Hemskärmar och appmenyer beskär ikoner till cirklar, rundade kvadrater och andra former, och en logotyp som fyller hela sin kvadrat förlorar sina kanter. Specifikationen definierar säkerhetszonen för en ikon med `maskable` som en cirkel i mitten med en radie på två femtedelar av ikonens kortare sida, och allt utanför den kan skäras bort. En ikon med `monochrome` behåller bara sin alfakanal, så en ikon som är ritad på en ogenomskinlig bakgrund blir en enfärgad kvadrat.

Färgfälten gör mindre än de ser ut att göra. `background_color` är bara tänkt för ögonblicket innan stilmallen har kommit fram, och specifikationen förbjuder webbläsaren att använda färgen när stilmallen väl finns tillgänglig. Det är färgen på den första bildrutan och ingenting efter den.

Så ser det ut i praktiken

Den här webbplatsen bygger sitt manifest ur en mall i stället för att hålla en fil för hand. Konfigurationen i Hugo registrerar `application/manifest+json` med ändelsen `webmanifest` och gör manifestet till en utdata från varje språks startsida, så varje språk får sin egen fil, och `head` på varje sida länkar till filen för sidans eget språk. Konfigurationen skriver också ned varför: en enda global kopia kan bara ange en startadress, och en dansk besökare som installerade webbplatsen fick en app som öppnade den engelska startsidan.

Mallen bygger manifestet som en datastruktur och lämnar den till JSON-kodaren i Hugo, så escapning och kommatecken är kodarens sak och ett saknat komma kan inte nå ut. Varje ikonadress utgår från webbplatsens rot, och kommentaren i mallen förklarar skälet: en relativ adress i `/da/manifest.webmanifest` skulle tolkas som en adress under `/da/`, och varje ikon på danska skulle ge 404. Själva

ikonerna läses från disk ur tre mappar, en per syfte, med tio storlekar i varje från 36 till 512 pixlar, och storleken tas ur filnamnet, så en storlek som generatorn lägger till dyker upp i manifestet utan att något behöver redigeras.

`id` är låst till språkets startadress, och `scope` är satt till roten, så hela webbplatsen ingår och en länk till ett annat språk stannar i det installerade fönstret. Skärmbilderna heter `<lang>-<form_factor>-<WxH>.webp`, och mallen läser alla tre uppgifterna ur filnamnet: engelska och danska har var sin breda bild på 1 280 × 800 pixlar och var sin smala bild på 780 × 1 688. Ett språk utan skärmbilder får inget `screenshots`-fält alls, i stället för en tom lista.

Kolofonen listar W3C-specifikationen bland de standarder som webbplatsen följer, och PWABuilder bland de externa kontroller som vem som helst kan köra: en tjänst som läser manifestet och resten av det som gör webbplatsen installerbar, från maskiner som Microsoft driver.

Sidan om att installera som app är läsarens halva av samma fil. Den går igenom var varje webbläsare har kommandot för att installera och noterar att webbläsarna från Apple aldrig erbjuder installation av sig själva. Firefox har en separat funktion för webbappfönster, aktiverad som standard på Windows från version 143, eller version 150 vid installation från Microsoft Store. Mozillas implementationsdokumentation anger att Linux-stödet är avstängt som standard och att stöd för macOS saknas. Sidan anger också förväntningarna på den här webbplatsen: den installerade appen är en ikon och ett eget fönster. Webbplatsen publicerar ingen service worker eller hanterad offlinekopia, så normal användning kräver en anslutning.

Var det går fel

Varje ikon ger 404 under en översatt sökväg. Ett relativt `src` tolkas relativt manifestets URL, så ett manifest som serveras under ett språkprefix ber om ikoner under det prefixet. Skriv ikonadresserna från webbplatsens rot, eller som fullständiga URL:er.

Den installerade appen öppnas på fel språk. Ett manifest kan bara ange en startadress, så varje läsare installerar samma startsida. Servera ett manifest per språk, vart och ett med egen `start_url` och eget `id`, och länka från varje sida till det manifest som hör till sidans språk.

Installerade kopior struntar i det nya manifestet. Utan `id` är identiteten startadressen, så en flyttad startsida ger filen en ny identitet, och webbläsaren bör behandla den som en annan app än den som läsarna redan har. Sätt `id` före den första versionen och låt det vara orört efteråt.

Hemskärmen beskär logotypen. En ikon vars `purpose` är `any` placeras eller beskärs som plattformen finner lämpligt. Lägg till en ikon med `maskable` och håll allt som betyder något innanför den centrala cirkeln, vars radie är 40 % av ikonens kortare sida.

Ändringar i manifestet ändrar ingenting. HTML-standarden använder bara det första `link` -elementet vars `rel` innehåller `manifest` , så när ett tema eller ett tillägg har lagt in en egen länk tidigare på sidan är filen som redigeras aldrig den som läses. Ha exakt en manifestlänk per sida.

Läs mer

- Web Application Manifest — varje fält, hur en webbläsare bearbetar filen och reglerna för `id` , `scope` och ikonernas syften.
- Web App Manifest - Application Information — `description` , `categories` och `screenshots` , med formfaktorer och etiketter.
- IANA: application/manifest+json — registreringen av medietypen och dess filändelse `.webmanifest` .
- HTML Standard: link type manifest — hur manifestlänken hämtas och varför bara den första räknas.
- Use web apps in Firefox for Windows — Windows-funktionen och dess tillgänglighet från version 143 eller Microsoft Store-version 150.
- Firefox Source Docs: Web Apps in Firefox — den aktuella implementationen: Windows aktiverat som standard, Linux avstängt som standard och macOS utan stöd.

Ordlista

Välj utifrån era faktiska förutsättningar

Beskriv arbetsflödet, befintliga gränssnitt och användarna som behöver förändringen. Vi kan definiera en testbar leverans och en överlämning som gör att teamet kan underhålla lösningen.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är ett webbappsmanifest?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

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

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/webmanifest/