The cross-browser future of Installable Web Apps

Metadata

IRC: `#InstallableWebApps`
Slides: 🟨 The cross-browser future of Installable Web Apps
Notes: bit.ly/notes-document (this document)

Present

- Thomas Steiner (**Chair**, Google Chrome)
- Diego Gonzalez-Zuniga (**Chair**, Microsoft Edge)
- Kenneth Christiansen (**Chair**, Chromium)
- Marcos Caceres (**Chair**, Apple Safari)
- Vincent Scheib (Google Chrome)
- Austin Sullivan (Google Chrome)
- Marian Harbach (Google Chrome)
- Michael Wilson (Google Chrome)
- Kagami Rosylight (Mozilla)
- Howard Wolosky (Microsoft Edge)
- Anssi Kostiainen (Intel)
- Andreas Bovens (Whereby)
- Lea Verou (Invited Expert)
- John Riviello (Comcast)
- Chris Lorenzo (Comcast)
- Christian Liebel (Thinktecture)
- Dan Murphy (Google Chrome)
- Penelope McLachlan (Google)
- Jonathan Kingston (DuckDuckGo)
- Tom Van Goethem (Google Chrome)
- Lu Huang (Microsoft Edge)
- Patrick Brosset (Microsoft Edge)
- Yves Lafon (W3C)
- Hakan Isbiliroglu (Google Chrome)
- Ali Spivak (Google Chrome)
- Johannes Wilm (Invited Expert)
- Ayu Ishii (Google Chrome)

Cursor parking

Discussion

- Diego: Side Bar Apps
  - Anssi: Is this side bar feature related to Edge proposal for [split tabs](#)?
  - Diego: Different
  - Howard: Correct
  - Howard: Those are unrelated
- Thomas: Can we expand more on widgets?
  - Diego: More like a companion app.
  - ???: Adaptive cards - you aren't writing web contents, you are writing description in adaptive card format, and a communication pipeline back and forth to modify etc.
    - Mechanism through the serviceworker

- Marcos: Installation Criteria
  - WebKit: Yes - any page can be installed.
  - Thomas: Chrome already doesn't require a service worker. We had to come up with a no-op SW mitigation, which is stupid. You could also always install any page as a shortcut with a standalone window.
  - Penny: Chrome allows install of any page, however pages that have intentionally presented name, icon, scope, etc are worth presenting to a user in a different way. E.g. 'any' web site may get minimal UI with back/forward, vs a web app with manifest that explicitly opts-in.
  - Anssi: What was removed exactly from the manifest spec?
  - Marcos: If the user wants something, even crappy, on their home screen, just do it.
  - Penny: Users should be allowed to do whatever they want.
  - Marcos: Frustration as an occasional Chrome user: couldn't install stuff, because some of the boxes weren't ticked.
  - Chris: Was pretty sad when the service worker req was removed. Was always telling developers they should provide a great offline experience. Like the fallback Chrome came up with to have default offline. Like that you get deduced icons and title, so fine to have no criteria. The greater value is in the same behavior regardless of the engine.
  - Vincent: Does anyone want to preserve criteria?
  - Dan: What about an icon as a strawman and an icon?
  - Marcos: No criteria at all.
  - Howard: Agreed – in Edge, the "Install Site as App" feature allows users to add their own icon / title if they so choose.
  - Penny: We want to keep the concept of promotability criteria.
  - Diego: From Edge, we're fine with having no criteria for install of any site. Still looking at the data, want to make sure Windows users have a good experience.
  - Marcos: Had consensus at the standards level, which is neat.
- Penelope McLachlan: Curious what Microsoft, Apple, Mozilla, others from the community believe the user problems solved by "installing" web apps is? Happy to share my perspective as well of course
  - Mozilla (Kagami): We don't have answers that we want to share.
  - Marcos: Adding to home screen moves from tab to OS level integration. Exposes benefits of e.g. notifications, home screen controls. Cohesive experience is important. Makes the transition easy. Safari has a very aggressive expulsion policy. Install gives a signal to lower the bar when it comes to expulsion so it happens less.
  - Penny: Integrations in OS level feature like run at startup, badging, are powerful. Some UXR means users can use apps installed and in tab at the same time. Users don't know how powerful the web is. When they install something, their mental model changes. Most users don't believe a tab can work offline until they experience it.

- Andreas Bovens (Whereby): How is scope defined for apps that are installed but don't have a manifest? At what point is the main web browser or a webview or custom tab opened (e.g. when going to a different domain or subdomain)? Are there differences between browsers?
    - Kenneth: If you don't have anything, it opens as a browser tab, at least in Chrome. Based on the start URL.
    - Dan (Chrome): The implied scope is wherever your page is, the directory of that page. That's how it's implemented in Chromium.
        - e.g. example.com/foo/page.html -> example.com/foo
    - Marcos: Same in Safari.
- Thomas: if any of the browser vendors have done user research, have they done research on a possible difference experience users have on apps being installed vs using in a tab.
    - Do they interact differently? Do they have a different mental model?
    - Anssi: There was a researcher in one of my working groups - big study. There was increased trust on things that look like apps. Learned behavior, users are trained.
    - Diego: Compared PWAs with native counterparts. Pretty much similar to what Anssi was saying. Window Controls Overlay, Web Share, etc. they didn't notice the difference. This was for apps obtained via browser (and not the Microsoft Store).
    - Penny: I can share similar findings. We've done UXR for same application in tab vs standalone window. Expectations is they could trust standalone more. also different expectations around capability of that app. Also interesting
        - Marian - we asked people how they make decisions about apps, what went into that. The source where they got it from ranked very highly. The follow-up exhaustively compared parts of the condition tree. How do you know, where you find it, installation ceremony, what does it look like after. Backlog to run this study - what matters.
            - Hypothesis is that it is very heavily frontloaded - trusting the source of the software.
            - Then, what the experience looks like after install is done.
    - Thomas: When it comes to source of the software - MSFT is going all-in for web apps in the MSFT store, PWA builder. Exact same experience you get in tab install vs MSFT store. Do you see confusion from people who obtain twitter via web app vs msft store, do they know how to differentiate? Is one "Just a website" and the other isn't? Do they know that the thing on the MSFT store is a web app?
        - Diego - the store does disclose that you need to have edge installed
        - not that I am aware of - we were constantly working on fixing little details that might 'give away' the app, the user kind of expects the concept of the polished 'native' app. We are constantly working on dressing small things, where something isn't appy, making it appy. Interesting challenge - you need to be very clear with the user that any change they are making on a permission is affecting the origin.

- - - Window controls overlay was a big one that bridged the gap allowing a modern title bar
    - in the end, user is using an application in the end.
- Thomas: Install experience & extensions
  - All browsers have extensions. Everywhere you can have them. Sometimes on both platforms
  - What does the user want, do they want them to run in installed experience? Or - is it bad enough that window controls overlay is too bad
  - Safari, install app is a little cleaner, no puzzle piece.
  - Is this something users would want in apps?
  - Kenneth: We need to ask users, power users probably. Maybe it could be hidden more, to make it less obvious.
  - Thomas: We had the puzzle piece hidden.
  - Dan: We had this for a while, but it was a security issue. I see extensions as a killer feature. It would be a disservice to users to not run extensions in apps.
  - Penny: Extensions are a web super power. Users can make the web as they like. Also accessibility argument to be made.
  - Howard: Heard feedback from developers who have PWAs in the Microsoft Store who don't like extensions in their store apps. Didn't like ads getting suppressed.
  - Penny: There are other examples apart from ads monetization: financial institutions who want to control more tightly. Or games where people could cheat with extensions. It's in the user's best interest, but the other argument does deserve recognition.
  - Marcos: Don't think we run extensions. (Confirmed after quick test.)
  - Johannes: If I have a text editor. Works well with Grammarly. Or actually Grammarly breaks my editor. It would be nice to allow apps to opt out of extensions.
    Marcos: It's ultimately a user decision. Would be better to bring this forward to the extension developer.
  - Ali: Working on extensions at Google. The conversation should happen between web apps & extensions groups. Extensions teams have concerns about allowing developers to disable extensions, for example sites who constantly ask for opt out of ad blocking.
  - Thomas: Maybe obtained via Store apps could have certain features enabled, like WCO. Might be making this up.
  - Kenneth: Do you get a different experience, or is the implementation exactly the same?
  - Howard: The experiences are pretty much the same. Given you install from the latest enough Win 11 or 10 version. They get the same set of tech.
  - Kenneth: You can't install twice?
  - Howard: You can, once from web, once from store. [NOTE: this statement was corrected below]
  - Kenneth: Why do they have different identities?
  - Howard: A store app gets a store identity. That's how the store works essentially.

- ○ Howard: (Post-session close update after double-checking) – I was mistaken and just double-checked this right after the session ended. If you install from the web, the Store will see it as already installed and will just offer to open it for you. Same the other way around. You shouldn't be able to install multiple instances of the same app via Edge/MS Store.