Browser extension

http://www.nathankrasney.com/

What is browser extension

- An extension adds features and functions to a browser.
- It's created using familiar web-based technologies — HTML, CSS, and JavaScript.
- Here are just a few examples of the things you can do (all are used by me):
  - Chrome dev tools
  - Grammarly
  - Lastpass
  - TubeBuddy
  - Waalaxy - #1 LinkedIn Automation Tool
  - Colorzilla

Motivation for browser extension

- It can take advantage of the same web APIs as JavaScript on a web page , but an extension also has access to its own set of JavaScript APIs. for chrome its Chrome API
- **You can do a lot more in an extension than you can with code in a web page** !!!! - example of useful stuff that can be done :
  - Create a browser tab using a predefined URL
- Make new products which are possible ONLY with web extensions
  - You can modify all existing websites on the world with browser extension - the possibilities and opportunities for new products - web extensions are giagentics. E.g. TubeBuddy add functionality to YouTube
  - You can create new products like Gramerlies and LastPass

What's the motivation of browser extension for me

I hope that it will be easy for automation in general and linkedin automation :

- send message - this requires authentication
- get post likes - this requires authentication ???
- get poll results - this requires authentication ???
- add welcome message - this requires authentication ???

Four types of files

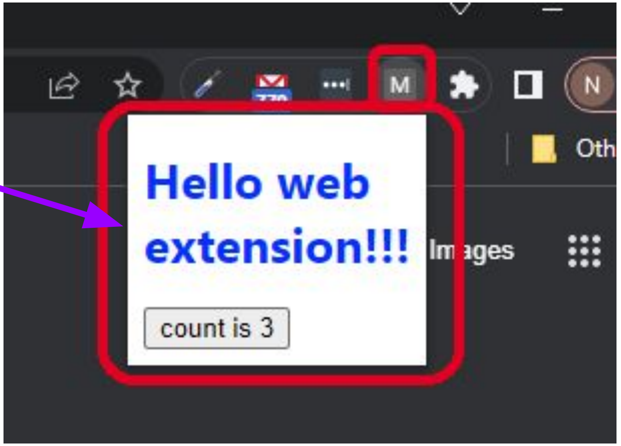
- The manifest
- Popup and other pages
- Content scripts
- The service worker

File type 1 : The manifest

- The extension's manifest is the only required file that **must** have a specific file name: manifest.json .
- It also has to be located in the extension's root directory.
- The manifest records important metadata, defines resources, declares permissions, and identifies which files to run in the background and on the page.
- Very simple example manifest.json (check also in tag 1.3 manifest.json)

File type 2 : Popup and other pages

- An extension can include various HTML files, such as a popup, an options page, and other HTML pages.
- All these pages have access to Chrome APIs
- Runs as long as the popup is run i.e. in the popup context
- For example index.html which is reference by "default_popup" : "index.html" in manifest.json



http://www.nathankrasney.com/

File type 3 : content scripts

- Content scripts execute Javascript in the context of a web page.
- They can also read and modify the DOM of the pages they're injected into.
- Content Scripts can only use a subset of the Chrome APIs, but can indirectly access the rest by exchanging messages with the extension service worker
- Runs as long as the web page run i.e. in the web page context

File type 4 : the service worker

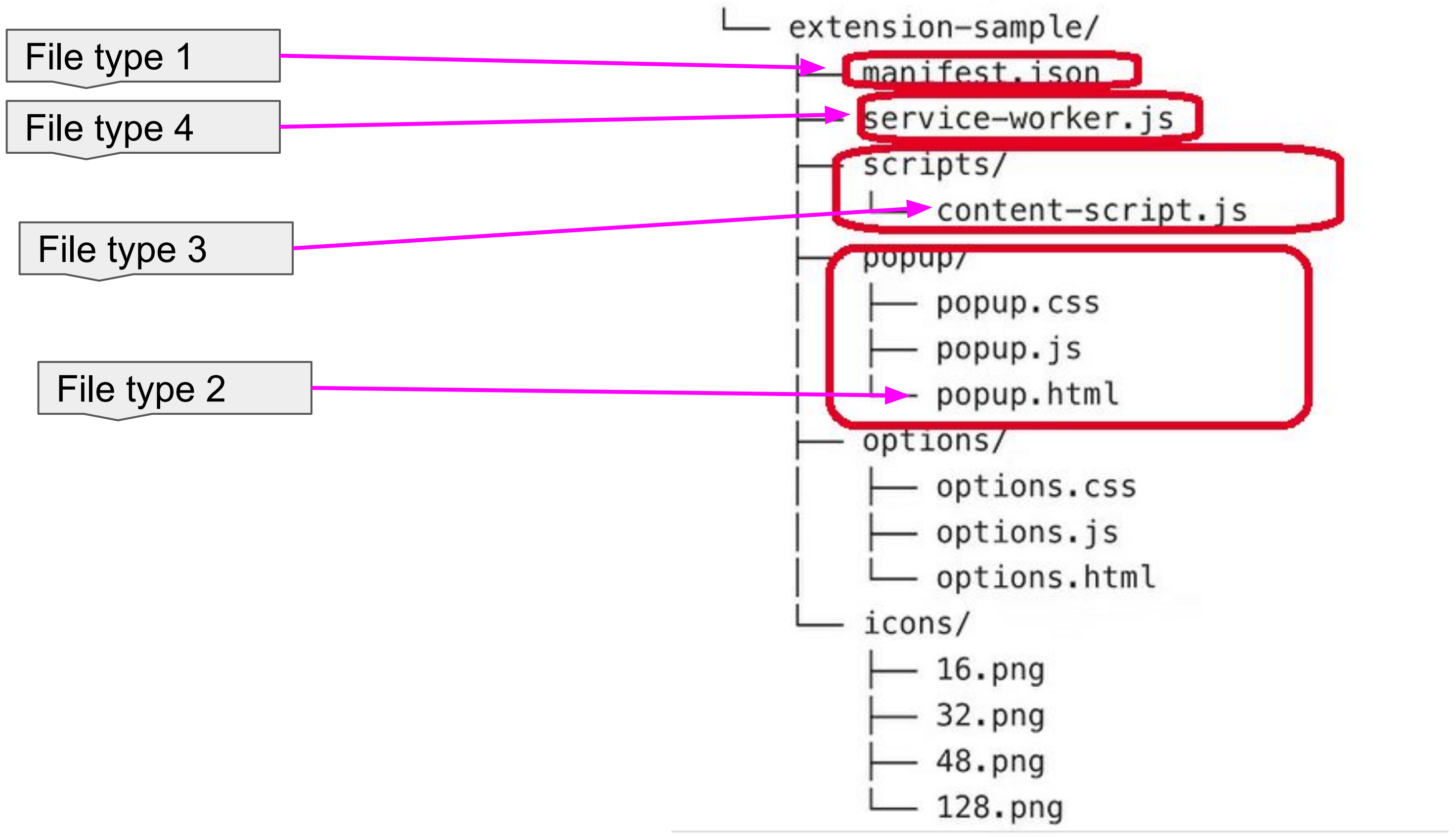
- The extension service worker handles and listens for browser events.
- There are many types of events, such as navigating to a new page, removing a bookmark, or closing a tab.
- It can use all the Chrome APIs, but it cannot interact directly with the content of web pages; that's the job of content scripts.
- This worker run separately from the main browser thread. So it it will not have access to web page, but do have ability to communicate check e.g. here
- Runs as long as the browser is run i.e. in the browser context

permissions

To use most Chrome APIs, your extension must declare its intent in the permissions fields of the manifest. Extensions can request four categories of permissions, specified using the respective keys in the manifest:

- permissions - Contains items from a list of known strings (such as "geolocation" , “tabs”).
- optional_permissions - Are like regular permissions, but are granted by the extension's user at runtime, rather than in advance.
- host_permissions - Contains one or more match patterns that give access to one or more hosts.
- optional_host_permissions Are like regular host_permissions, but are granted by the extension's user at runtime, rather than in advance.

Suggested project structure



```
└── extension-sample/
    ├── manifest.json
    ├── service-worker.js
    ├── scripts/
    │   └── content-script.js
    ├── popup/
    │   ├── popup.css
    │   ├── popup.js
    │   └── popup.html
    ├── options/
    │   ├── options.css
    │   ├── options.js
    │   └── options.html
    └── icons/
        ├── 16.png
        ├── 32.png
        ├── 48.png
        └── 128.png
```

http://www.nathankrasney.com/

Nice google chrome examples

Hello extension - display “Hello Extensions” when the user clicks on the extension toolbar icon. Very similar to my sample here

reading time - To insert an element on every page automatically. Github project is here. Its uses

- icons
- content scripts to inject code into pages
- match patterns
- extension permissions

My repositories

browser-extension-simple-vanilla-js-plaground - most minimal plugin including install

Browser-extension-vanilla-js-plaground - use content script and icon

Web extension next.js - for production using api

Difference between browser extension in firefox vs chrome

while browser extensions serve similar purposes in Firefox and Chrome, there are differences in terms of APIs, marketplaces, compatibility, openness, privacy, and development environments.

Developers may need to consider these differences when building extensions for specific browsers or ensuring cross-browser compatibility.

Users should also be aware of potential differences in extension availability and functionality when using different browsers.

webextension-polyfill

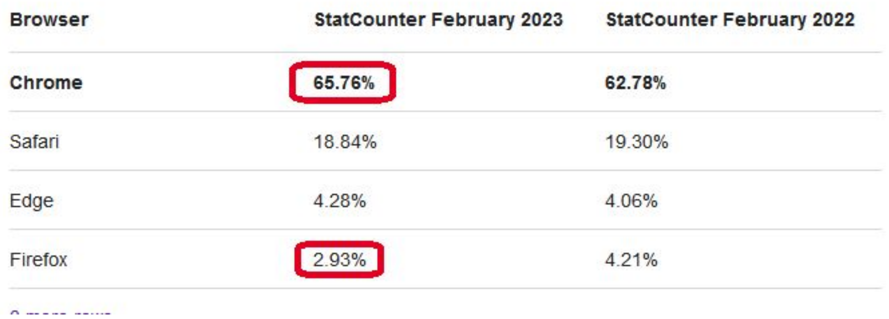
Cross browser web extension api used e.g. here

chrome or firefox as browsers

Key Takeaways:

- Chrome and Firefox are close to being even in most of their capabilities.
- Chrome is faster and has a larger library of extensions, but Firefox is more private and secure.
- Firefox is fast, but suffers from inefficient RAM consumption.

As of 2023 the market share



Browser	StatCounter February 2023	StatCounter February 2022
Chrome	**65.76%**	**62.78%**
Safari	18.84%	19.30%
Edge	4.28%	4.06%
Firefox	2.93%	4.21%

Comparison link

http://www.nathankrasney.com/

Develop first browser extension

Facts :

1. I like firefox
2. Chrome popularity is rising while firefox popularity declines
3. chrome is x 20 time popular as of feb 2023
4. I might sell the extension thus market share is important

Decision - first extension will be developed for chrome

References 1

Learn about developing extensions for Chrome. - very nice and easy to use

development-basics - chrome docs - very nice and easy to use

Chrome-extensions-samples - official google docs samples

Creating a Chrome extension with React and TypeScript - aug 2021

CODE WITH ME: Build a Chrome Extension | How to Build & Publish a Chrome Extension in 10 Minutes? - oct 2022

Build a Chrome Extension – Course for Beginners - May 2022 , github project is here . it use : chrome api storage , content script , service worker (not sure its needed) , update the youtube dom , video element , popup html

References 2

What is Background Script / Service Worker | Chrome Extension 101 | Video 07 | TUTORIEX - nov 2022

Mdn - browser extensions - excellent firefox docs

Webextensions-examples - excellent firefox samples