Pake - wrapping websites

Walking the internet site by site is fun but sometimes you might want to jump back to a specific site you've been before - like weather forecast page. That's what bookmarks are for or history search. But what if you want that page to open up in a new window? You can easily move the browser tab and make a new browser window with that one tab. But what if you want to take it further or even faster? What if the page you jump to could pop up in a new window automatically just like your other system applications? That's where Pake enters the game.

No, not another Electron promoting article. Since I hate Electron so much I don't even talk about that anymore that would be silly from me. The solution I'm thinking about is to simply wrap a webpage to Tauri shell. That's exactly what Pake does.

Once your webpage is packed with Pake you can open it up as a standalone application. You can interact with the page as it's open in your browser thus you can sign in to your email or another service. App data are persistent so no repetitive signing in. You can create a tray icon (which I personally like very much). You can preset size window etc. The are is browser UI window bloat - just the page. There are many options that you can use and more are even coming.

The build process in not that simple. You have to use Node.js and since Pake is written in Rust then the whole Rust installation. Also a few system libs are required which are in the time of writing this post quite broken. I assume that's because there is new 2.0 Tauri version which is not quite finished yet + libwebkit libs and it's dependencies are broken in Debian. So to install the whole build environment is not that easy but there is another much cleaner solution.

You can simply build the app in a docker container and then discard the whole container. If you build the app as App Image then you get much bigger app (~100MB instead of ~5MB) but it contains the whole story. No system packages are required anymore. In times like these where system dependencies are not 100% in sync with what you need it's a lifesaver.

So how to execute?

```
 $ docker run --rm -it ivangabriele/tauri:debian-bookworm-18 bash
```

That downloads the container and spawns bash inside.

```
$ npm i -g pake-cli
```

That installs Pake.

```
$ pake maps.google.com --targets appimage
```

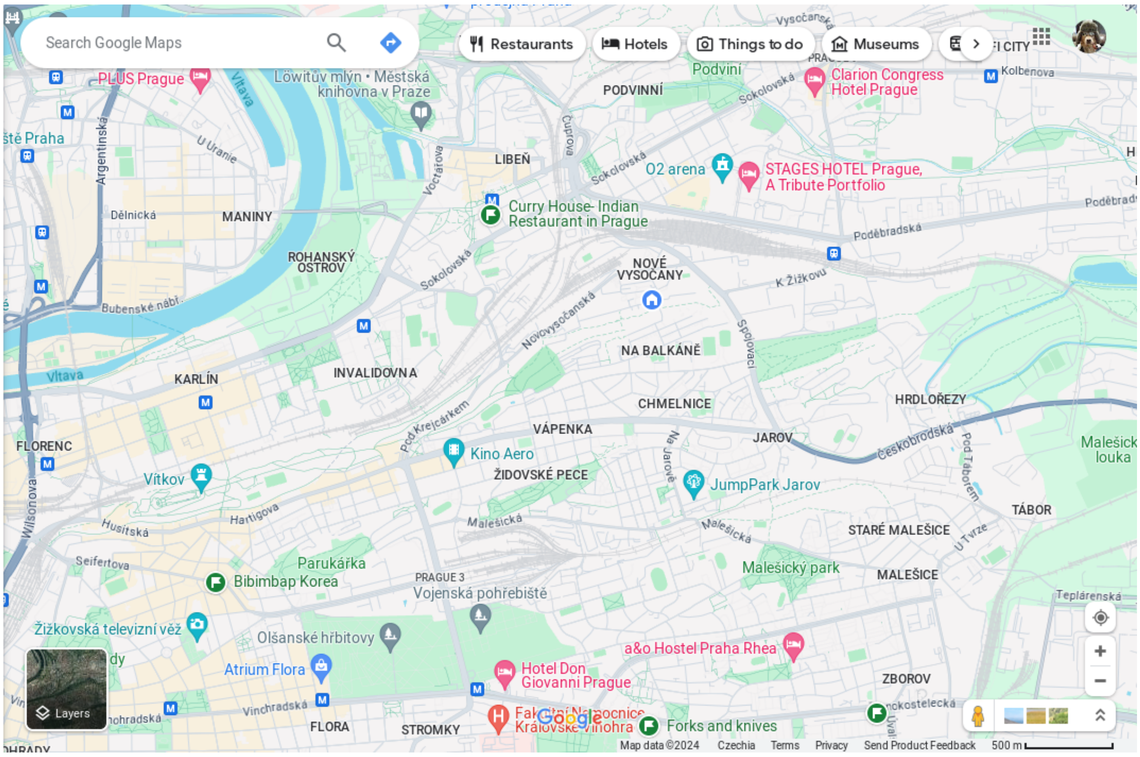
And that wraps Google Maps into a standalone application. Now you just need to copy the built application over to your local machine (out of the container) and you are good to go.

```
$ docker copy CONTAINER:SOURCE ~/bin/maps
```

Assuming your local `~/bin` directory is in your PATH you can run the file/command

```
$ maps
```

And this is what you get



The applications opens in like ~1s on my 7 years old i7 CPU and when idling it takes no CPU at all.