🧰

SDK Overview

Our first version of the SDK is for the JavaScript ecosystem. Therefore, users can build web apps, backends (Node.js and Deno) and native apps (React Native and Capacitor.js) with our SDK. We have the widest reach by choosing a JavaScript SDK first. In addition, we also develop our SDK with TypeScript, which means that the type declarations for developers using TypeScript instead of JavaScript are already built in.

Our SDK contains different libraries that can work independently of each other. This is a modern approach to building libraries, as people already treeshake a big portion of their packages by installing only sub-parts of an SDK they need, rather than installing the whole SDK, which already contains all libraries, even if they wouldn't use them (ergo dead and redundant code).

Nonetheless, we still use a monorepo for all the libraries of the SDK and provide each version only through the main versions of the SDK in order not to clutter our GitHub organisation with hundreds of repositories. Take a look at the NPM package registry in our GitHub organisation to see all the libraries of the IDUN Guardian SDK.

If you visit the https://github.com/iduntech/idn-guardian-sdks repository, you will see that we create releases that trigger a CI/CD pipeline to deliver patch, minor and major versions of the SDK. We follow the practice of semantic versioning to tag our SDK via Git.

How we document our SDK

We will continue to document our SDK in a dedicated repository on GitHub, which will turn out to be a third party website that can also be used as documentation.
More information will follow soon. @Daniel Burger