

Conclusion



By Ivan Akulov

Contracting software engineer · Blogging, open source, performance, UX

Summing up:

- **Cut unnecessary bytes.** Compress everything, strip unused code, be wise when adding dependencies
- **Split code by routes.** Load only what's really necessary right now and lazy-load other stuff later
- **Cache code.** Some parts of your app are updated less often than other ones. Separate these parts into files so that they are only re-downloaded when necessary
- **Keep track of the size.** Use tools like [webpack-dashboard](#) and [webpack-bundle-analyzer](#) to stay aware how large is your app. Take a fresh look at your app's performance at whole every few months

Webpack is not the only tool that could help you make an app faster. Consider making your application a [Progressive Web App](#) for even better experience and use automated profiling tools like [Lighthouse](#) to get improvement suggestions.

Don't forget to read [webpack docs](#) – they have plenty of other useful information.

And make sure to play [with the training app](#)!

[Previous](#)

← [Monitor and analyze the app](#)

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated July 2, 2018.