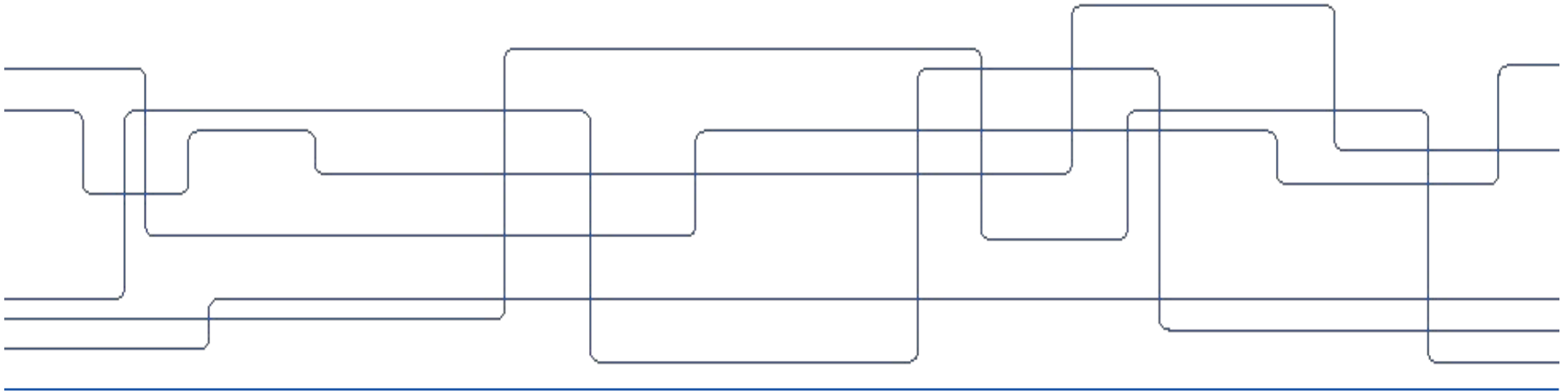# Open source culture and conventions

Automated Software Testing and DevOps
Julian Nalenz
May 10th, 2022

Structure

- goal of this presentation: give inspiration for your own (open source) projects on which conventions to follow so that others feel comfortable using it
    - Initial README.md
    - Documenting code
    - Semantic versioning
    - Changelogs
    - Commit messages
    - Branch naming
    - Automated building/unit testing
    - Licenses
- further reading: https://opensource.guide

Initial README.md

- first entry point → good and trustworthy impression!

- properly learn Markdown!

- common sections

    - Short description

    - How to run the project/examples (related: INSTALL.md)

    - Link to further documentation

    - Contributing instructions (related: CODE_OF_CONDUCT.md, CONTRIBUTING.md)

    - Active maintainers (related: OWNERS.md)

    - License (related: LICENSE)

- examples: React, VLC, Spring Framework, Android (Dalvik)

Documenting code

- popular standards: Javadoc (Oracle), Doxygen

- separate documentation page using static site generator, hosted for free e.g. on GitHub Pages

  - https://reactnative.dev based on Docusaurus (specifically focused on documenting Open Source projects; write Markdown and get out HTML!)

  - https://reactjs.org based on Gatsby

- https://www.i18next.com based on GitBook (free only for open source)

- documenting APIs: Swagger/OpenAPI
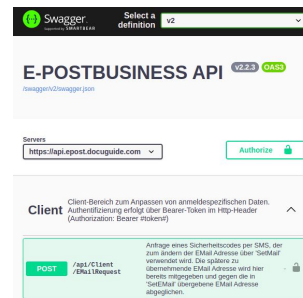

```
/**
 * Has an alert occurred?
 *
 * <p>
 * This function never throws an exception
 * </p>
 *
 * @return true if there is an alert
 */
boolean isAlertPresent();
```

https://github.com/SeleniumHQ/selenium/blob/trunk/java/src/com/thoughtworks/selenium/Selenium.java#L827-L836

https://api.epost.docuguide.com/swagger/index.html

Semantic versioning

- main purpose: ensuring backwards compatibility, allow users of your source to determine immediately whether upgrading might break something

- https://semver.org: Given a version number MAJOR.MINOR.PATCH, increment the:

  – MAJOR version when you make incompatible API changes,

  – MINOR version when you add functionality in a backwards compatible manner, and

  – PATCH version when you make backwards compatible bug fixes.

- very important e.g. for NPM, because `npm update` does not automatically upgrade to new major versions

Changelogs

- to be found in **CHANGELOG.md**

- specify release date, version number, breaking changes, notable changes, new features, bug fixes, deprecations

- link to associated pull requests and mention contributors

- if needed, link migration guide

18.0.0 (March 29, 2022)

Below is a list of all new features, APIs, deprecations, and breaking changes. Read React 18 release post and React 18 upgrade guide for more information.

New Features

React

- `useId` is a new hook for generating unique IDs on both the client and server, while avoiding hydration mismatches. It is primarily useful for component libraries integrating with accessibility APIs that require unique IDs. This solves an issue that already exists in React 17 and below, but it's even more important in React 18 because of how the new streaming server renderer delivers HTML out-of-order.
- ...

Deprecations

- `react-dom`: `ReactDOM.render` has been deprecated. Using it will warn and run your app in React 17 mode.
- `react-dom`: `ReactDOM.hydrate` has been deprecated. Using it will warn and run your app in React 17 mode.
- `react-dom`: `ReactDOM.unmountComponentAtNode` has been deprecated.
- `react-dom`: `ReactDOM.renderSubtreeIntoContainer` has been deprecated.
- `react-dom/server`: `ReactDOMServer.renderToNodeStream` has been deprecated.

Breaking Changes

https://github.com/facebook/react/blob/main/CHANGELOG.md

8.219.0 - 2022-04-21

- #1398 API Updates
  - Add support for `expire` test helper method on resource `Refund`
  - Change type of `BillingPortal.Configuration.application` from `string` to `expandable($Application)`
  - Change `IssuingDisputeCreateParams.transaction` to be optional

8.218.0 - 2022-04-18

- #1396 API Updates
  - Add support for new resources `FundingInstructions` and `Terminal.Configuration`

https://github.com/stripe/stripe-node/blob/master/CHANGELOG.md

Commit messages

- feat : (new feature for the user, not a new feature for build script)
- fix : (bug fix for the user, not a fix to a build script)
- docs : (changes to the documentation)
- style : (formatting, missing semi colons, etc; no production code change)
- refactor : (refactoring production code, eg. renaming a variable)
- test : (adding missing tests, refactoring tests; no production code change)
- chore : (updating grunt tasks etc; no production code change)

https://gist.github.com/joshbuchea/6f47e86d2510bce28f8e7f42ae84c716

- Conventional Commits

- or commit message prefixes according to affected module

The commit message should be structured as follows:

```
<type>[optional scope]: <description>

[optional body]

[optional footer(s)]
```

https://www.conventionalcommits.org/en/v1.0.0/

Commits on May 5, 2022

test: rename handlewrap.hasref tests ···
daeyeon committed 1 hour ago

bootstrap: stop delaying instantiation of maps in per-context scripts ···
RaisinTen committed 3 hours ago

test: improve observable ICU behaviour coverage ···
LiviaMedeiros committed 3 hours ago

test: validate webstream encoder/decoder inspector ···
kuriyosh committed 3 hours ago ✗

test: use mustSucceed instead of mustCall with assert.ifError ···
fossamagna committed 3 hours ago ✗

test: improve lib/internal/webstreams/readablestream.js coverage ···
fossamagna committed 3 hours ago ✓

url: revert "validate ipv4 part length" ···
aduh95 committed 3 hours ago ✓

bootstrap: use a context snapshotted with primordials in workers ···
joyeecheung committed 3 hours ago ✓

deps: upgrade npm to 8.9.0 ···
npm-robot committed 5 hours ago ✗

https://github.com/nodejs/node/commits/master

29 Apr, 2022 20 commits

contrib: main.mak: force LIBDIR to lib/ ···
Alexandre Janniaux authored 1 week ago and Rémi Denis-Courmont committed 5 days ago

vdpau/interop: fix non-reentrant pointers (fixes #26554)
Rémi Denis-Courmont authored 1 week ago

qt: fix typos ···
Romain Vimont authored 1 week ago and Rémi Denis-Courmont committed 5 days ago

qt: fix typo in filename ···
Romain Vimont authored 1 week ago and Rémi Denis-Courmont committed 5 days ago

qt: handle error messages separately from dialog ···
Pierre Lamot authored 1 week ago and Rémi Denis-Courmont committed 5 days ago

dialogs: split callbacks between interactive and non-interactive callbacks ···
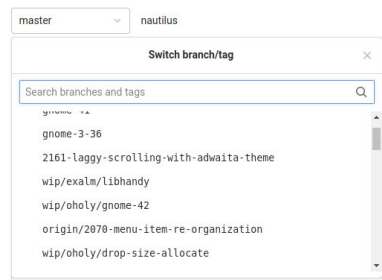Pierre Lamot authored 1 week ago and Rémi Denis-Courmont committed 5 days ago

qt: fix QtQuick version ···
Romain Vimont authored 1 week ago and Pierre Lamot committed 6 days ago

contrib: regex: Apply a patch to add missing function declarations ···
Martin Storsjö authored 1 week ago

https://code.videolan.org/videolan/vlc/-/commits/master
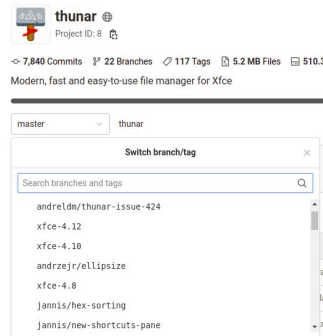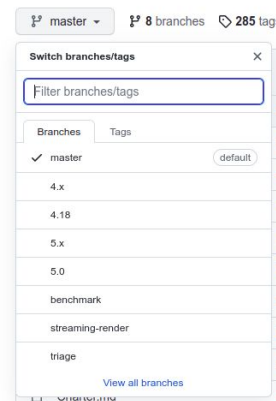
Branch naming

- possibility for feature/bugfix branches: include the following

  - username of contributor

  - related issue number

  - issue title in a few words

  - "wip", "bug" or "devel" prefix

- often one branch for each released version

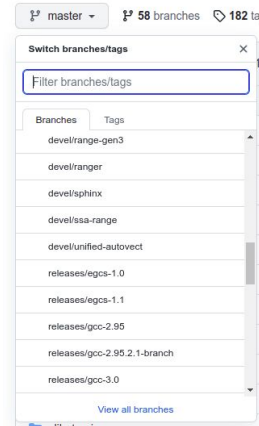- not super relevant on GitHub or GitLab as repositories are commonly forked

https://gitlab.gnome.org/GNOME/nautilus

https://gitlab.xfce.org/xfce/thunar

https://github.com/expressjs/express

https://github.com/gcc-mirror/gcc

Automated building/unit testing

- executing unit tests or building binaries automatically using GitHub actions and show status in readme → more trust

- more tests (e.g. fuzzing, code quality, test coverage, linting) also found occasionally

- popular providers: GitHub Actions, TravisCI, CircleCI, Azure DevOps, …
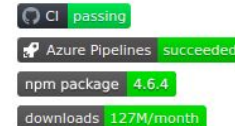
- badges for example from https://shields.io

https://github.com/curl/curl

https://github.com/nodejs/node
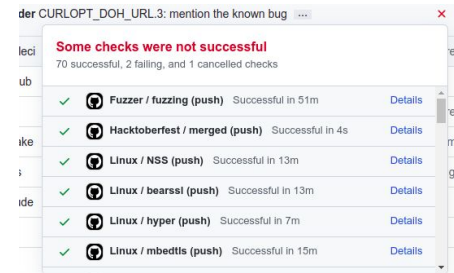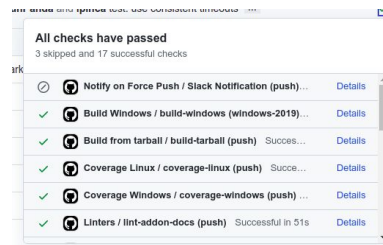
https://github.com/microsoft/TypeScript

Licenses

- common licenses (also see https://choosealicense.com)

  - MIT, Apache: commercial use, distribution, modification, private use

  - GNU GPL: **must** disclosing source and use same license

- use SPDX License Identifier (e.g. "MIT" or "GPL-3.0-or-later") to make chosen license unambiguous

- well-known environments

  - all around Linux: mostly GNU GPL

  - Node.js/NPM: mostly MIT, Apache etc.

Take-home message

Consistently follow conventions to increase productivity of everyone!

Questions?