

DEC 2022



# GitHub Fundamentals

**Lukasz Dynowski**

[linkedin.com/in/ldynia](https://www.linkedin.com/in/ldynia)



# API Styles Fundamentals

Published by [O'Reilly Media, Inc.](#)

## Understanding trade-offs and choosing the correct solution for your architecture

[What you'll learn](#) [Is this live event for you?](#) [Schedule](#)

### In this course, you'll:

- Learn how a variety of APIs are designed
- Understand the differences among API communication methods
- Learn which API design decisions are best for which use cases

### Course description

If you only have a hammer, everything begins to look like a nail, but sticking with a single API style is limiting. Nowadays, a service or microservice has several APIs, and they must all be well-defined.

Join expert Lukasz Dynowski to broaden your API style palette. You'll examine the best types of APIs for your use cases as well as the design decisions you must make for RESTful APIs, GraphQL APIs, Websockets API, gRPC API, and messaging protocols. You'll learn how to design a variety of APIs, recognize the trade-offs among them, and understand where to use each one for the best communication among applications.

Feb. 28, 2023

2 - 5 p.m. Central European Standard Time

112 Spots Remaining

March 23, 2023

5 - 8 p.m. Central European Standard Time

174 Spots Remaining

[Sign up for a free trial!](#)

or [sign in](#).

### YOUR INSTRUCTOR



Lukasz Dynowski

**Lukasz Dynowski** is an independent consultant who in his career was involved in over 150+ projects. Counting over 10 years of experience as a software engineer Lukasz was doing Full-Stack, DevOps, Software Architecture, as well as...

[Read more](#)





# Objectives

GitHub Repository	CI/CD
GitHub Codespaces	Docker
GitHub Pull Requests	Git
GitHub Templates	Markdown
GitHub Releases / Tag	Microservices
GitHub Issues	Free Software
GitHub Actions	Open Source
GitHub Wikis	GitHub is your CV
GitHub Pages	
GitHub Advanced Search	





# GitHub in Numbers



94+ million Developers



200+ million Repositories



3+ million Organizations



90% Fortune 500





# How did we arrived here?

Unix	<b>1970</b>	AT&T Bell Labs
GNU	<b>1983</b>	GNU Project
GCC	<b>1987</b>	GCC + Libs
GPL	<b>1989</b>	4 Freedoms
Linux Kernel	<b>1991</b>	Linux Kernel
Open Source	<b>1998</b>	Netscape
GIT	<b>2005</b>	BitKeeper
GitHub	<b>2008</b>	Logical Awesome LLC
	<b>2021</b>	





# Free Software & Open Source



**Richard Stallman**



**Bruce Perens**



**Eric Raymond**







# What does “free software” mean?

## User has freedom to:

- **Run** the program as you wish, for any purpose.
- **Study** how the program works and change it, so it does your computing as you wish.
- **Redistribute** copies so you can help others.
- **Distribute** copies of your modified versions to others.

Access to the source code is a precondition for this!

<https://www.gnu.org/philosophy/free-sw.en.html>





# What does “**open source**” mean?

The software is:

- **Redistributed** for free.
- The source code should be **publicly available**.
- The software can be **modified and distributed** in a different format from the original software.
- The software should **not discriminate against persons or groups**.
- The software should **not restrict the usage of other software**.

<https://opensource.org/osd>







# Free software **vs** Open source

## Free Software

Social movement.

Free software focus on the user's freedom to use the program, to modify it, and to share it.



## Open source

Open source is a development methodology.

Open-source focus on the availability of the source code and the ability to modify and share it.





# Is Public repository open source?

Public **≠** Open Source

Anyone can view and fork your public project, but your work comes with no permissions.

**What if you don't apply an open source license?**

**Everybody** who contributes to your project also **becomes an exclusive copyright holder of their work**. That means **nobody can use, copy, distribute, or modify their contributions** – and that “nobody” includes you.

If you want others to use, distribute, modify, or contribute back to your project, you need to include an open source license.





# What is a software license?

An open source license is a legal contract between the author of the software and the user of a software. An open source license states what the user is permitted to do with the software, their obligations, and what they cannot do.





# What license to choose from?

- 0-clause BSD License (0BSD)
- 1-clause BSD License (BSD-1-Clause)
- 2-clause BSD License (BSD-2-Clause)
- 3-clause BSD License (BSD-3-Clause) (<https://choosealicense.com/se>)
- Academic Free License 3.0 (AFL-3.0)
- Adaptive Public License (APL-1.0)
- Apache Software License 1.1 (Apache-1.1) (superseded)
- Apache License 2.0 (Apache-2.0)
- Apple Public Source License (APSL-2.0)
- Artistic license 1.0 (Artistic-1.0) (superseded)
- Artistic License 2.0 (Artistic-2.0)
- Attribution Assurance License (AAL)
- Boost Software License (BSL-1.0)
- BSD License: See
  - 3-clause BSD License
  - 2-clause BSD License
  - 1-clause BSD License
  - 0-clause BSD license
  - BSD-3-Clause-LBNL
- BSD+Patent (BSD-2-Clause-Patent)
- CERN Open Hardware Licence Version 2 - Permissive
- CERN Open Hardware Licence Version 2 - Weakly Reciprocal
- CERN Open Hardware Licence Version 2 - Strongly Reciprocal
- CeCILL License 2.1 (CECILL-2.1)
- Common Development and Distribution License 1.0 (CDDL-1.0)
- Common Public Attribution License 1.0 (CPAL-1.0)
- Common Public License 1.0 (CPL-1.0) (superseded)
- Computer Associates Trusted Open Source License 1.1 (CATOSL-1.1)
- Cryptographic Autonomy License v.1.0 (CAL-1.0)
- CUA Office Public License Version 1.0 (CUA-OPL-1.0) (retired)
- Eclipse Public License 1.0 (EPL-1.0) (superseded)
- Etc.

200+

- Etc.
- Open Software License 3.0 (OSL-3.0)
- OpenLDAP Public License Version 2.8 (OLDAP-2.8)
- OSET Public License version 2.1
- PHP License 3.0 (PHP-3.0) (superseded)
- PHP License 3.01 (PHP-3.01)
- The PostgreSQL License (PostgreSQL)
- Python License (Python-2.0) (overall Python license)
- CNRI Python license (CNRI-Python) (CNRI portion of Python License)
- Q Public License (QPL-1.0)
- RealNetworks Public Source License V1.0 (RPSL-1.0)
- Reciprocal Public License, version 1.1 (RPL-1.1) (superseded)
- Reciprocal Public License 1.5 (RPL-1.5)
- Ricoh Source Code Public License (RSCPL)
- Sleepycat Open Font License 1.1 (OFL-1.1)
- Simple Public License 2.0 (SimPL-2.0)
- Sleepycat License (Sleepycat)
- Sun Industry Standards Source License (SISSL) (retired)
- Sun Public License 1.0 (SPL-1.0)
- Sybase Open Watcom Public License 1.0 (Watcom-1.0)
- Universal Permissive License (UPL)
- University of Illinois/NCSA Open Source License (NCSA)
- Upstream Compatibility License v1.0
- Unicode Data Files and Software License
- The Unlicense
- Vovida Software License v. 1.0 (VSL-1.0)
- W3C License (W3C)
- wxWindows Library License (WXwindows)
- X.Net License (Xnet)
- Zero-Clause BSD (0BSD)
- Zope Public License 2.0 (ZPL-2.0)
- zlib/libpng license (Zlib)





# What is a software license?

	Public Domain	Permissive	Copyleft	Noncommercial license	Proprietary	Trade secret
Rules	Grants all rights	Grants use rights, including right to <b>relicense</b>	Grants use rights, forbids <b>proprietyization</b>	Grants rights for noncommercial use only.	No rights granted	No information made public
Example	CC*, WTFPL	Apache, MIT, BSD	GPL, AGPL	JRL, AFPL	Copyrights, no public license	
Users	< 1%	Kubernetes, GitLab	Linux Kernel, Gimp			





# What license to choose from?

**<https://choosealicense.com>**

<https://blog.usejournal.com/understanding-open-source-and-free-software-licensing-c0fa600106c9>





# Change the license of my project?

## CHOOSE WISELY !

<https://opensource.guide/legal/#what-if-i-want-to-change-the-license-of-my-project>

