



Curated List of Best DevOps Tools

devopscube.com/best-devops-tools/

☆ 527 stars 🍴 187 forks 👁 12 watching 🔑 1 Branch 🏷 0 Tags 🔄 Activity

🌐 Public repository

 1 Branch 🏷 0 Tags 🔑 🏷

Go to file 

Go to file

+

Add file ▾

Code

...

 techiescamp Update README.md

ca4f08b · 3 months ago 

 CONTRIBUTING.md	List Update	last year
 README.md	Update README.md	3 months ago

Curated List of DevOps Tools

A comprehensive collection of essential DevOps tools for development, deployment, monitoring, and more.

✍ Feel free to enhance the list of devops tools by submitting pull requests! Read the [contribution guidelines](#) to know more.

Hit the Star! ★

If you are planning to use this repo for reference, please hit the star. Thanks!

Table of Contents

- [Development Environment Tools](#)
- [Source Code Management](#)
- [Build Tools](#)
- [Continuous Integration Tools](#)
- [Artifact Management Tools](#)
- [Code Analysis Tools](#)
- [Continuous Delivery & GitOps Tools](#)
- [Infrastructure Provisioning Tools](#)
- [Cloud Cost Management Tools](#)
- [Configuration Management Tools](#)
- [Secret Management Tools](#)
- [Config/Service Discovery Tools](#)

- [Containerization Tools](#)
- [Container Orchestration Tools](#)
- [Container Security Tools](#)
- [Policy Management Tools](#)
- [Service Mesh Tools](#)
- [Logging Tools](#)
- [Monitoring & Observability Tools](#)
- [Visualization Tools](#)
- [Internal Developer Platform Tools](#)
- [API Tools](#)
- [Collaboration Tools](#)
- [Backups and Restoration Tools](#)
- [Cloud Providers](#)

Development Environment Tools

Development environment tools are essential for maintaining consistency in software configurations and versions. They ensure reproducibility in minutes in case of a crash, increasing developer productivity.

- [Virtual Box](#) (Open Source): A powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use.
- [Qemu](#) (Open Source): A generic and open source machine emulator and virtualizer.
- [Vagrant](#) (Open Source): A tool for building and managing virtual machine environments in a single workflow.
- [Docker Desktop](#) (Open Source): An application for MacOS and Windows machines for the building and sharing of containerized applications.
- [Minikube](#) (Open Source): A tool that makes it easy to run Kubernetes locally.
- [Minishift](#) (Open Source): A tool that helps you run OpenShift locally by running a single-node OpenShift cluster inside a VM.
- [Podman Desktop](#) (Open Source): A daemonless container engine for developing, managing, and running OCI Containers on your Linux System.
- [Rancher Desktop](#) (Open Source): Rancher Desktop is a tool that enables developers to manage and deploy Kubernetes clusters effortlessly on their local machines.
- [kind](#) (Open Source): kind is a tool that uses Docker containers as nodes to facilitate the creation of local Kubernetes cluster primarily used for local development and testing.
- [k3d](#) (Open Source): k3d is a lightweight wrapper to run k3s in Docker and makes it easy to create single and multi-node k3s clusters.

Source Code Management

In the DevOps world, everything is treated as code (e.g., code, shell script, configurations, etc.). Robust source code management tools are a must.

- [Github](#) (Free & Paid): A web-based hosting service for version control using Git. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features.
- [Bitbucket](#) (Free & Paid): A web-based version control repository hosting service owned by Atlassian, for source code and development projects that use either Mercurial or Git revision control systems.

- [Gitlab](#) (Free & Paid): A web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking and CI/CD pipeline features, using an open-source license.
- [AWS CodeCommit](#) (Free & Paid): A fully-managed source control service that makes it easy for companies hosting their own repositories to collaborate on code in a secure and highly scalable ecosystem.
- [Google Cloud Source Repositories](#) (Free & Paid): Fully-featured, scalable, private Git repositories hosted on Google Cloud.
- [Azure Repos](#) (Free & Paid): Provides Git repositories or Team Foundation Version Control (TFVC) for source control of your code.
- [Codeberg](#) (Free & Paid): Cloud-based instance of Forgejo.
- [Forgejo](#) (Open Source): Open-source fork of Gitea.
- [Fossil](#) (Free): Open-source distributed SCM that includes a wiki, issue management, and other features. Not compatible with git.
- [Gitea](#) (Open Source, Free, Paid Support Available): Open-source fork of Gogs.
- [Gogs](#) (Open Source): Open-source git-based SCM, loosely based on Github's UI.

Build Tools

Build tools are essential for automating package creation or deployable artifacts.

- [Maven](#) (Java) (Open Source): A software project management and comprehension tool, primarily used for Java projects.
- [Gradle](#) (Java, Kotlin, Groovy, Scala, Python, C++) (Free & Paid): An open-source build automation system that builds upon the concepts of Apache Ant and Apache Maven.
- [npm](#) (Javascript): A package manager for JavaScript, used to install, share, and distribute code.
- [pnpm](#) (Open Source): A fast, disk space efficient package manager for JavaScript.
- [yarn](#) (Open Source): A fast, reliable, and secure dependency manager for JavaScript and TypeScript.
- [Rake](#) (Ruby) (Open Source): A software task management and build automation tool for Ruby.
- [MSBuild](#) (.Net) (Open Source): The build platform for .NET and Visual Studio.
- [Pybuilder](#) (Python) (Open Source): A software build tool written in pure Python mainly for Python applications.

Continuous Integration Tools

Continuous Integration provides continuous feedback on code integrations, helping solve issues faster and decreasing time in software release cycles.

- [Jenkins](#) (Open Source): An open-source automation server, helps to automate parts of the build, test, and deployment process. Java-based, with Groovy-based pipeline definition.
- [Github Actions](#) (Free & Enterprise): A CI/CD solution that integrates with GitHub repositories to run a series of commands automatically.
- [Gitlab CI](#) (Free & Enterprise): A continuous integration service included with GitLab that builds and tests the software whenever the developer pushes code to the application. YAML-based pipelines. Can be self-hosted or use the cloud offering.
- [CircleCI](#) (Free & Paid): YAML-based CI/CD hosted in the cloud.
- [Drone](#) (Free & Paid): YAML-based CI/CD that is container-first. Can self-host or use the cloud version.
- [TeamCity](#) (Free & Enterprise): A build management and continuous integration server from JetBrains.
- [Travis CI](#) (Open Source): A cloud-based continuous integration service that automatically builds and tests code changes in GitHub repositories.

- [Bamboo](#) (Free & Enterprise): A continuous integration and deployment tool that ties automated builds, tests, and releases together in a single workflow.
- [RazorOps CICD](#) (Free & Paid): YAML-based CI/CD that is container-first SaaS cloud version and On-Premis with large Enterprise.
- [Buildkite](#) (Free & Paid): YAML-based CI/CD.
- [DeployBot](#) (Free & Paid): A cloud-based CI/CD service that supports building, deploying and automating any project.

Artifact Management Tools

Artifact management tools store and version code/binary that has to be deployed into production. They centralize repositories specifically made for this purpose.

- [Nexus](#) (Free & Enterprise): A repository manager that allows you to proxy, collect, and manage your dependencies.
- [Jfrog Artifactory](#) (Enterprise): An enterprise-grade universal binary repository manager solution that could handle end-to-end artifact management.
- [npm](#) (Free): A package manager for JavaScript, used to install, share, and distribute code.
- [Nuget](#) (.Net) (Free): A free and open-source package manager for the Microsoft development platform.

Code Analysis Tools

Code analysis tools are a crucial component of any DevOps toolkit. They scrutinize the code before deployment, pinpointing potential issues. This allows developers to enhance the quality of the software by making necessary modifications.

Code analysis can be categorized into two types:

- **Static Analysis (SAST)** – This method inspects the source code without executing it, identifying bugs, security vulnerabilities, and quality issues.
- **Dynamic Analysis (DAST)** – This approach involves running the code to uncover performance bottlenecks and behavioral anomalies.

Here are some notable tools in this category:

- [Sonarqube](#) (Free & Enterprise): SonarQube offers a comprehensive suite for continuous code quality inspection. It automates reviews with static code analysis to detect bugs, code smells, and security vulnerabilities.
- [Checkmarx](#) (Enterprise): Checkmarx is a leading solution in software security, adept at identifying, tracking, and rectifying both technical and logical security flaws in the source code.
- [PMD](#) (Open Source): PMD is a versatile static code analyzer. It can detect a wide range of programming flaws across multiple languages.
- [Acunetix](#) (Enterprise): Acunetix is a proprietary web vulnerability scanner capable of autonomously crawling and scanning web applications to identify diverse security concerns. It comes under DAST tools.

Continuous Delivery & GitOps Tools

Continuous Delivery is more of an approach rather than a toolset, aiming to ensure that software can be reliably and immediately deployed at any time. While there's no specific tool labeled solely for continuous delivery, several process-oriented tools can be leveraged to achieve this goal within your projects.

GitOps, a subset of DevOps, uses Git as a single source of truth for declarative infrastructure and applications. Here are some prominent tools that align with Continuous Delivery and GitOps principles:

- [Jenkins](#) (Open Source): A widely-used open-source automation server that supports building, deploying, and automating any project.
- [Argo CD](#) (Open Source): A declarative, GitOps continuous delivery tool for Kubernetes.
- [Flux CD](#) (Open Source): A tool that automates the deployment of containers to Kubernetes, following the GitOps principles.
- [Go CD](#) (Open Source): An open-source continuous delivery server designed to model and visualize complex workflows.
- [Gitlab CD](#) (Free & Enterprise): GitLab's continuous delivery solution, allowing automated deployment and monitoring of applications.
- [Weave GitOps](#) (Enterprise): An enterprise-grade GitOps solution, providing a set of best practices for deploying and managing applications and infrastructure.
- [Jenkins X](#) (Open Source): An open-source project that provides automated CI/CD for Kubernetes, with preview environments and promotion across environments.
- [Tekton](#) (Open Source): A powerful and flexible open-source framework for creating CI/CD systems, allowing developers to build, test, and deploy across cloud providers and on-premises systems.

Infrastructure Provisioning Tools

Infrastructure provisioning tools play a vital role in automating the setup of computing infrastructure. This includes the creation and management of virtual machines, networks, storage, and various cloud resources. By automating these processes, these tools enable more efficient and consistent infrastructure deployment.

Here are some of the leading tools in this category:

- [Terraform](#) (Open Source & Enterprise): Terraform is an open-source tool that allows you to define and provision a datacenter infrastructure using a declarative configuration language. It supports various cloud providers and offers an enterprise version with additional features.
- [Pulumi](#) (Open Source & Enterprise): Pulumi provides a way to create, deploy, and manage infrastructure using programming languages you already know. It offers both open-source and enterprise solutions.
- [CloudFormation](#) (AWS Service): An AWS service that helps you model and set up Amazon Web Services resources. It allows you to use a template to describe all the AWS resources you need, automating the provisioning process.
- [Azure Resource Manager](#) (Azure Service): Azure Resource Manager enables you to work with the resources in your solution as a group. You can deploy, update, or delete all the resources for your solution in a single, coordinated operation.
- [Bicep](#) (Azure Service): Azure Bicep is a simple and user-friendly domain-specific language (DSL) for describing Azure cloud resources. It helps to create, deploy, and manage Azure resources more easily and efficiently compared to traditional ARM templates. Bicep simplifies the process by offering a cleaner and more concise syntax, making it quicker to write and maintain Azure infrastructure code.
- [opentofu](#) (Open Source): OpenTofu is an infrastructure as code tool that lets you define both cloud and on-prem resources in human-readable configuration files that you can version, reuse, and share. You can then use a consistent workflow to provision and manage all of your infrastructure throughout its lifecycle. OpenTofu can manage low-level components like compute, storage, and networking resources, as well as high-level components like DNS entries and SaaS features.

Backups and Restoration Tools

Backup and restoration tools help in making periodic copies of data and applications to a separate, secondary device and then using those copies to recover the data and applications.

- [Velero](#) (Open Source): Velero is an open source tool to safely backup and restore, perform disaster recovery, and migrate Kubernetes cluster resources and persistent volumes.
- [Kasten K10](#) (Free & Enterprise): Kasten K10 is a data management platform that provides robust Kubernetes backup and disaster recovery solutions for seamless containerized application deployment and data protection.
- [Cloudcasa](#) (Free & Enterprise): CloudCasa is a cloud-native backup and disaster recovery solution designed to protect and manage data for Kubernetes applications seamlessly.

Cloud Cost Management Tools

Cloud Cost Management Tools help organizations monitor, analyze, and optimize their cloud spending. By providing insights into resource utilization and cost allocation, these tools enable better budgeting and cost control. Here's a list of tools that can be used for Cloud Cost Management:

- [InfraCost](#) (Open Source): InfraCost provides cost estimates for Terraform projects, helping developers understand the cost implications of their infrastructure code.
- [Kubecost](#) (Open Source): Kubecost gives you visibility into your Kubernetes usage and cost, allowing you to monitor and optimize your cluster resources.

Configuration Management Tools

Configuration management tools are pivotal in both application and infrastructure deployment. They assist in managing the state of servers and applications, ensuring that configurations are applied consistently and idempotently. By automating these configurations, these tools enhance the reliability and efficiency of deployment processes.

Here's a look at some of the widely-used configuration management tools:

- [Ansible](#) (Open Source & Enterprise): Ansible is a versatile automation tool that can handle configuration management, application deployment, and task automation. It uses a simple, human-readable language, allowing for easy adoption.
- [Chef](#) (Open Source & Enterprise): Chef is a powerful automation platform that transforms complex infrastructure into code, automating how infrastructure is configured, deployed, and managed across your network.
- [Puppet](#) (Open Source & Enterprise): Puppet is designed to manage the configuration of Unix-like and Microsoft Windows systems declaratively. It provides a way to automate repetitive tasks, quickly deploy critical applications, and proactively manage infrastructure.
- [Saltstack](#) (Open Source & Enterprise): SaltStack is an intelligent IT automation platform that can manage, secure, and optimize any infrastructure. It's built on a unique and responsive remote execution engine, allowing for control over thousands of systems with no performance degradation.

Secret Management Tools

Secret management tools are essential for safeguarding sensitive information such as passwords, API keys, and other credentials. They provide a secure way to store, distribute, and manage secrets, ensuring that they are accessible only to authorized entities.

Here are some popular secret management tools:

- [HashiCorp Vault](#) (Open Source & Enterprise): Vault is a tool for securely accessing secrets such as API keys, passwords, or certificates. It provides a unified interface to any secret while providing tight access control and recording a detailed audit log.
- [External Secrets Operator](#) (Open Source): An extension for Kubernetes that allows you to securely manage secrets stored in managed cloud services
- [AWS Secrets Manager](#): A service that helps you protect access to your applications, services, and IT resources without the upfront investment and on-going maintenance costs of operating your own infrastructure.
- [Google Cloud Secret Manager](#): A fully managed service on Google Cloud Platform to handle sensitive data like API keys, passwords, and certificates. It provides robust security and convenient access controls.
- [Azure Key Vault](#): A cloud service provided by Microsoft to securely manage keys, secrets, and certificates.
- [Teller](#) (Open Source): Teller is a secret management tool for developers enhancing productivity by supporting cloud-native applications and managing secrets across multiple cloud providers.

Config/Service Discovery Tools

Config and Service Discovery Tools are vital in distributed systems, allowing applications to manage configuration data dynamically and discover services without hard-coded hostnames or ports. These tools provide a centralized repository for configuration and service information, ensuring consistency and availability.

Here are some popular tools in this category:

- [Consul](#) (Open Source & Enterprise): Consul is a tool for discovering and configuring services in your infrastructure. It provides key/value storage and multi-datacenter support, ensuring high availability and scalability.
- [Etcd](#) (Open Source): A distributed key-value store that provides a reliable way to store data across a cluster of machines. It's primarily used in distributed systems to hold configuration data that needs to be available to all nodes in the cluster.
- [Apache ZooKeeper](#) (Open Source): A centralized service for maintaining configuration information, naming, providing distributed synchronization, and group services.
- [Eureka](#) (Open Source): A REST-based service that is primarily used in the AWS cloud for locating services for the purpose of load balancing and failover of middle-tier servers.

Containerization Tools

Containerization Tools enable the encapsulation of an application and its dependencies into a 'container.' This ensures that the application runs the same way, regardless of where it's deployed. By isolating the application in a container, these tools provide a consistent environment that's abstracted from the host system.

Here are some popular tools in this category:

- [Docker](#) (Open Source & Enterprise): Docker is a platform that enables developers to create, deploy, and run applications in containers. It's widely used for its ease of use and integration with various orchestration tools.
- [Podman](#) (Open Source): Podman is a daemonless container engine for developing, managing, and running OCI Containers on your Linux System. It provides a Docker-CLI-compatible command-line front

end and can use many of the same commands.

- [Buildah](#) (Open Source): Buildah is a tool that facilitates building OCI container images. It's a complementary tool to Podman and allows for more fine-grained control over image creation and management.
- [rkt](#) (Open Source): An application container engine developed for modern production cloud-native environments, emphasizing simplicity, security, and composability.
- [CRI-O](#) (Open Source): CRI-O is a lightweight container runtime specifically for Kubernetes. It allows Kubernetes to use any OCI-compliant runtime as the container runtime for running pods. It provides an optimized performance for Kubernetes workloads and is known for its simplicity and reliability.

Container Orchestration Tools

With the widespread adoption of containerization technologies like Docker, Container Orchestration Tools have become indispensable in microservices-based deployments. These tools manage the lifecycle of containers within clusters, automating tasks such as deployment, scaling, networking, and availability.

Here are some of the leading tools in this category:

- [Kubernetes](#) (Open Source): Kubernetes is the de facto standard in container orchestration. It automates the deployment, scaling, and management of containerized applications across clusters of hosts.
- [OpenShift](#) (Open Source & Enterprise): OpenShift is a Kubernetes-based container platform that provides developer and operational tools. It extends Kubernetes by adding features such as a web console, monitoring, logging, and more.
- [Nomad](#) (Open Source & Enterprise): Nomad is a flexible and easy-to-use orchestrator to deploy and manage containers and non-containerized applications. It integrates seamlessly with popular DevOps tools and provides a lightweight solution for container orchestration.
- [k3s](#) (Open Source): k3s is a certified Kubernetes distribution which is highly available and designed for resource-constrained production workloads.

Container Security Tools

- [Docker Bench](#) (Open Source): A script that checks for common best-practices around deploying Docker containers in production.
- [Clair](#) (Open Source): A vulnerability analysis tool for containers that inspects containers before they are deployed.
- [Anchore Engine](#) (Open Source): A service that analyzes and inspects containers for security vulnerabilities and policy issues.
- [AquaSec](#) (Enterprise): A container security platform that provides full lifecycle security for containerized applications.
- [Notary](#) (Open Source): A project that allows anyone to have trust over arbitrary collections of data.
- [Falco](#) (Open Source): A behavioral activity monitor designed to detect anomalous activity in applications.
- [Trivy](#) (Open Source): A simple and comprehensive vulnerability scanner for containers.
- [Twistlock](#) (Enterprise): Twistlock is an innovator in container security. During runtime, Twistlock scans container images for known vulnerabilities in the software packages and libraries they contain. It's new name is Prisma Cloud.

Policy Management Tools

Policy Management Tools enable organizations to define, enforce, and manage security policies across their cloud-native environments. They provide a framework for policy definition and evaluation, ensuring compliance with organizational and regulatory requirements.

- [Open Policy Agent](#) (Open Source): A general-purpose policy engine that unifies policy enforcement across the stack.
- [Kyverno](#) (Open Source): A policy engine designed for Kubernetes, providing validation, mutation, and generation of configurations.
- [Cloud Custodian](#) (Open Source): A rules engine for managing public cloud accounts and resources.

Service Mesh Tools

Service Mesh Tools provide a dedicated infrastructure layer for handling service-to-service communication. They make it easy to build and deploy resilient, secure, and observable microservices.

- [Istio](#) (Open Source): An open platform to connect, manage, and secure microservices, providing a uniform way to secure, connect, and monitor microservices.
- [Linkerd](#) (Open Source & Enterprise): A service mesh that gives you observability, reliability, and security without requiring any code changes.
- [Cilium Service Mesh](#): A next-generation service mesh for transparently securing, observing, and connecting Kubernetes workloads across complex, heterogeneous environments.

Logging Tools

For any production environment, logging and monitoring for infrastructure & application components are a must-have to avoid system failures and application downtime. Here is the list of tools you could use for your projects:

- [ELK Stack](#) (Open Source & Enterprise): A combination of Elasticsearch, Logstash, and Kibana, used for searching, analyzing, and visualizing log data in real-time.
- [Grafana Loki](#) (Open Source and Enterprise): A horizontally scalable, highly available, multi-tenant log aggregation system inspired by Prometheus.
- [Fluentd](#) (Open Source): An open-source data collector for unified logging layers.
- [Graylog](#) (Open Source & Enterprise): A leading centralized log management solution for capturing, storing, and enabling real-time analysis of terabytes of machine data.
- [logz.io](#) (Enterprise): A cloud-native logging and security platform that enables engineers to use the best open-source tools in the market without the complexity of operating them.
- [Splunk](#) (Enterprise): A platform for searching, monitoring, and examining machine-generated big data.
- [Syslog-ng](#) (Enterprise): A flexible and highly scalable logging solution enabling you to centralize your log data and sort it in real-time.

Monitoring & Observability Tools

Monitoring & Observability Tools provide insights into the performance, availability, and overall health of applications, infrastructure, and business processes. They enable proactive monitoring, alerting, and troubleshooting, ensuring optimal performance and reliability.

- [Prometheus](#) (Open Source): An open-source monitoring and alerting toolkit designed for reliability and scalability.

- [Thanos](#) (Highly Available Prometheus Open Source Tool): A set of components that can be composed into a highly available metric system with unlimited storage capacity.
- [Cilium](#) (Open Source & Enterprise): eBPF-based Networking, Security, and Observability.
- [Falco](#) (Open Source & Enterprise): Cloud Native Runtime Security.
- [Calico](#) (Open Source & Enterprise): Pluggable eBPF-based networking and security for containers and Kubernetes.
- [Sensu](#) (Open Source & Enterprise): A comprehensive monitoring solution for your entire infrastructure and application stack.
- [Riemann](#) (Open Source): A powerful stream processing language that helps you compose, test, and distribute data streams.
- [Nagios](#) (Free & Enterprise): A powerful monitoring system that enables organizations to identify and resolve IT infrastructure problems.
- [Zabbix](#) (Open Source & Enterprise): An enterprise-class open-source distributed monitoring solution.
- [Middleware](#) (Free & Enterprise): A full-stack observability platform that offers a free developer account with all features and a paid plan for higher limits/usage.
- [Data Dog](#) (Enterprise): A monitoring and analytics platform for large-scale applications.
- [New Relic](#) (Enterprise): A cloud-based platform that gives developers, engineers, operations, and management a clear view of what's happening in today's complex software environments.
- [App Dynamics](#) (Enterprise): An application performance management solution that uses machine learning and artificial intelligence to provide real-time visibility and insight into IT environments.
- [Sumologic](#) (Enterprise): A cloud-native, machine data analytics platform that delivers real-time, continuous intelligence.
- [Dynatrace](#) (Enterprise): An AI-powered, full-stack, automated performance management solution.

Visualization Tools

- [Grafana](#) (Open Source & Enterprise): A leading open-source platform for monitoring and observability, allowing you to query, visualize, alert on, and understand your metrics.
- [Kibana](#) (Open Source & Enterprise): A free and open user interface that lets you visualize your Elasticsearch data and navigate the Elastic Stack.
- [Tableau](#) (Enterprise): A leading data visualization and business intelligence platform that helps people see and understand their data.

Internal Developer Platform Tools

Platform engineering is becoming a core part of every organization that wants to adopt DevOps. Internal developer platforms or IDP tools play a key part in platform engineering. The following are the key tools that can be used for IDP:

- [Backstage.io by Spotify](#) (Open Source): An open platform for building developer portals, providing a unified frontend for all your infrastructure tooling, services, and documentation.
- [Port.io](#) (Free & Enterprise): A platform that enables teams to build, test, and deploy applications with ease.
- [Configure8](#) (Free & Paid): A platform that provides a unified interface for managing infrastructure as code.
- [Cortex](#) (Enterprise): An enterprise-grade platform that simplifies the deployment, scaling, and management of machine learning models.
- [Opslevel](#) (Enterprise): A platform that provides insights and automation for your microservices and cloud resources.

API Tools

- [Postman](#) (Free & Enterprise): A collaboration platform for API development.
- [Hoppscotch](#) (Open Source): Open-source API development Ecosystem.
- [SoapUI](#) (Open Source & Enterprise): The world's leading automated testing tool for SOAP and REST APIs.
- [Swagger](#) (Open Source & Enterprise): A framework for API specification that includes a suite of tools for auto-generating documentation, code generation, and API testing.
- [HTTPIe](#) (Open Source): HTTPIe (pronounced aitch-tee-tee-pie) is a command-line HTTP client. Its goal is to make CLI interaction with web services as human-friendly as possible

Collaboration Tools

The following are the collaboration tools that you could use in your DevOps toolchain. These tools support both messaging and video calling:

- [Slack](#) (Free & Paid): A collaboration hub that connects your work with the people you work with.
- [Cisco Webex Teams](#) (Free & Paid): A collaboration app that helps you create, meet, message, call, whiteboard, and share, regardless of whether you're together or apart.
- [Flock](#) (Free & Paid): A communication and collaboration platform designed to boost productivity and foster teamwork.
- [Google Hangouts](#) (Free & Paid): A communication platform that includes messaging, video chat, and VOIP features.
- [Flow Dock](#) (Paid): A team collaboration app for desktop, mobile, and web, working seamlessly alongside your existing tools.

Planning & Project Management Tools

Following are the planning tools that you could use in your DevOps toolchain:

- [Jira](#) (Free/Paid): A popular project management tool for agile teams.
- [Trello](#) (Free/Paid): A collaboration tool that organizes your projects into boards.
- [Asana](#) (Free/Paid): A web and mobile application designed to help teams organize, track, and manage their work.
- [Backlog.com](#) (Free/Paid): A project management and collaboration tool for teams that want higher productivity.
- [Monday.com](#) (Paid): A work operating system that powers teams to run projects and workflows with confidence.
- [ClickUp](#) (Free/Paid): A cloud-based productivity and project management platform that supports

📖 README

✎ ☰

IDE Tools

- [Visual Studio Code](#) (Free): A free source-code editor made by Microsoft.
- [Sublime Editor](#) (Free): A sophisticated text editor for code, markup, and prose.
- [Notepad++](#) (Free): A free source code editor and Notepad replacement.

Bug/Issue Tracking Tools

Bug and issue-tracking tools are a must for any team dealing with code. Here is the list of tools you might want to have a look at:

- [Backlog](#) (Free & Paid): A project management and collaboration tool with bug tracking capabilities.
- [Bugzilla](#) (Open Source): A server software designed to help you manage software development.
- [Jira](#) (Free & Paid): Also used for bug tracking, workflow, and issue tracking.
- [Lean Testing](#) (Free): A free bug tracking and test case management platform.
- [Mantis](#) (Free): A free web-based bug tracking system.

Test Automation/Performance Testing Tools

Another important pillar in the CI/CD pipeline is test automation tools. The following are the automation/performance testing tools that top our list:

- [Selenium](#): A suite of tools to automate web browsers.
- [UFT](#): Unified Functional Testing tool for automated functional testing.
- [Appium](#): An open-source tool for automating native, mobile web, and hybrid applications.
- [Jmeter](#): An open-source software to test performance both on static and dynamic resources.
- [Blazemeter](#): A continuous testing platform for performance testing, monitoring, and more.
- [Tosca](#): A software testing tool that leverages the Tosca standard. Tricentis Tosca is an enterprise-level, continuous testing platform that uses model-based test automation to enable accelerated testing.

Centralized Documentation Management tools

Documentation is an essential aspect of the DevOps culture, facilitating effective collaboration, knowledge management, and overall efficiency in the software development and delivery process. Here is the list of tools you might want to have a look at:

- [Confluence](#) (Free & Paid): A collaborative team workspace and wiki-based content management system developed by Atlassian. It is designed to facilitate team collaboration, knowledge sharing, and project documentation within organizations.
- [Clickup](#) (Paid): A cloud-based productivity and project management platform that supports document management. It helps teams collaborate, organize tasks, and streamline workflows, making it easy to store, organize, and collaborate on various documents and files.

Cloud Providers

Cloud service provider is an information technology (IT) company that provides its customers with computing resources over the internet and delivers them on-demand. CSPs are well-suited for organizations and individuals who don't want the responsibility of installing software, hardware or network resources — and maintaining them until the end of their life cycles.

No releases published

Packages

No packages published

Contributors 12

