

# **GALILEO PLATFORM**

**CREAZIONE**                      26/02/22  
**AGGIORNAMENTO**            09/01/23  
**PARTECIPANTI**              ITAmmb ( Italia Massimiliano Buscati )

## **ATTORI LATO CLIENT**

- 1. MOZILLA FIREFOX**
- 2. JAVASCRIPT**

## **ATTORI TRASFERIMENTO DATI**

- 3. W3C**
- 4. AJAX**
- 5. XML**
- 6. JSON**

## **ATTORI LATO SERVER**

- 1. APACHE HTTP SERVER**
- 2. APACHE TOMCAT**
- 3. APACHE STRUTS**
- 4. SPRING**
- 5. PHP**
- 6. LARAVEL**

## **LINGUAGGIO DI PROGRAMMAZIONE**

- 1. JAVA PLATFORM - STANDARD EDITION**
- 2. JAKARTA EE**
- 3. JAVA SERVER PAGES**
- 4. SERVLET**
- 5. DESIGN PATTERN: MVC**

## **RANKING**

- 1. TIOBE PROGRAMMING COMMUNITY INDEX**
- 2. COMMON VULNERABILITIES AND EXPOSURES**

# ATTORI LATO CLIENT



Genere	<a href="#">Browser</a>
Sviluppatore	Mozilla Corporation e <a href="#">Mozilla Foundation</a>
Data prima versione	23 settembre 2002
Ultima versione	<ul style="list-style-type: none"><li>91.6.0esr (8 febbraio 2022)</li><li>97.0.1 (17 febbraio 2022)</li></ul>
Ultima <a href="#">beta</a>	98.0beta (8 febbraio 2022)
Ultima <a href="#">alfa</a>	99.0a1 (7 febbraio 2022)
<a href="#">Sistema operativo</a>	<a href="#">CSS</a> <a href="#">Rust</a> <a href="#">XUL</a> <a href="#">JavaScript</a> <a href="#">C</a> <a href="#">C++</a>
<a href="#">Linguaggio</a>	
<a href="#">Toolkit</a>	<a href="#">XML User Interface Language</a> e <a href="#">GTK</a>
<a href="#">Licenza</a>	<a href="#">MPL 2.0</a> ( <a href="#">licenza libera</a> )
<a href="#">Lingua</a>	Multilingua (90 lingue[1])
<a href="#">Sito web</a>	<a href="http://www.mozilla.org/firefox/new/">www.mozilla.org/firefox/new/</a>

**Mozilla Firefox** (conosciuto semplicemente come **Firefox**) è un [web browser libero](#) e [multiplatforma](#), mantenuto da [Mozilla Foundation](#).<sup>[2]</sup> Nato nel [2002](#) con il nome "Phoenix" dai membri della comunità Mozilla, i quali volevano un browser [stand-alone](#) piuttosto che il raggruppamento [Mozilla Application Suite](#), fino alla versione 56 ha usato il [motore di rendering Gecko](#), supportando gran parte dei nuovi standard web oltre ad alcune caratteristiche che sono state progettate come estensioni a questi ultimi; dalla versione 57, *Gecko* è stato sostituito da [Quantum](#).

La sua diffusione raggiunse il picco massimo nel 2009 con il 32% dell'utenza,<sup>[3]</sup> divenendo con la versione 3.5 il browser più utilizzato<sup>[4][5]</sup>; da quel momento iniziò un lungo declino nel numero di utenti a causa della concorrenza di [Google Chrome](#)<sup>[3]</sup>; nel dicembre [2014](#) Mozilla annunciò che il browser aveva raggiunto cinquecento milioni di utenti.<sup>[6]</sup> Nel gennaio [2016](#), la diffusione mondiale, come browser per [PC desktop](#), si attestava tra il 9% e il 16%, rendendolo così il secondo browser più diffuso<sup>[7][8][9][10]</sup>.

Firefox è nato nel settembre [2002](#) come una sperimentazione interna del gruppo di sviluppo di Mozilla. Ai tempi, con il progetto ancora finanziato dalla [Netscape Communication Corporation](#), l'autonomia decisionale dei programmatori riguardo alle scelte di fondo del progetto era in qualche modo limitata e non era inusuale che si cercasse una maggiore libertà di azione in piccoli progetti non ufficiali. [Dave Hyatt](#) (ora responsabile del browser [Safari](#)) e [Blake Ross](#) diedero dunque vita a Firefox come sperimentazione sull'usabilità e per dimostrare come fosse possibile adottare un approccio diverso nella produzione di un browser targato Mozilla.<sup>[24]</sup>

[ [https://it.wikipedia.org/wiki/Mozilla\\_Firefox](https://it.wikipedia.org/wiki/Mozilla_Firefox) ]



Autore	<a href="#">Brendan Eich</a>
Data di origine	<a href="#">1995</a>
Ultima versione	1.8.5

**JavaScript** è un [linguaggio di programmazione](#) multi paradigma [orientato](#) agli [eventi](#), comunemente utilizzato nella [programmazione Web lato client](#) (esteso poi anche al [lato server](#)) per la creazione, in [siti web](#) e [applicazioni web](#), di effetti dinamici [interattivi](#) tramite [funzioni](#) di [script](#) invocate da *eventi* innescati a loro volta in vari modi dall'utente sulla [pagina web](#) in uso ([mouse](#), [tastiera](#), caricamento della pagina ecc...).

Originariamente sviluppato da [Brendan Eich](#) della [Netscape Communications](#) con il nome di **Mochan** e successivamente di **LiveScript**, in seguito è stato rinominato "*JavaScript*" ed è stato formalizzato con una [sintassi](#) più vicina a quella del linguaggio [Java](#) di [Sun Microsystems](#) (che nel 2010 è stata acquistata da [Oracle](#)). [Standardizzato](#) per la prima volta il 1997 dalla [ECMA](#) con il nome **ECMAScript**, l'ultimo standard, di giugno 2021, è ECMA-262 Edition 12<sup>[1]</sup> ed è anche uno standard [ISO](#) (ISO/IEC 16262)

[ <https://it.wikipedia.org/wiki/JavaScript> ]

# ATTORI TRASFERIMENTO DATI



HyperText Transfer Protocol (HTTP) (in italiano: protocollo di trasferimento di un ipertesto) è un protocollo a livello applicativo usato come principale sistema per la trasmissione d'informazioni sul web ovvero in un'architettura tipica client-server. Le specifiche del protocollo sono gestite dal World Wide Web Consortium (W3C). Un server HTTP generalmente resta in ascolto delle richieste dei client sulla porta 80 usando il protocollo TCP a livello di trasporto.

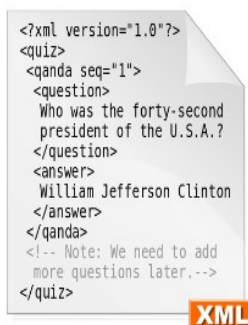
[ [https://it.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](https://it.wikipedia.org/wiki/Hypertext_Transfer_Protocol) ]

**AJAX**, acronimo di **Asynchronous JavaScript and XML**, è una tecnica di sviluppo software per la realizzazione di applicazioni web interattive (Rich Internet Application), basandosi su uno scambio di dati in background fra web browser e server, consentendo così l'aggiornamento dinamico di una pagina web senza esplicito ricaricamento da parte dell'utente.

La tecnica AJAX utilizza una combinazione di:

- HTML (o XHTML) e CSS per il markup e lo stile;
- DOM (Document Object Model) manipolato attraverso un linguaggio ECMAScript come JavaScript o JScript per mostrare le informazioni ed interagirvi;
- l'oggetto XMLHttpRequest per l'interscambio asincrono dei dati tra il browser dell'utente e il web server. In alcuni framework AJAX e in certe situazioni, può essere usato un oggetto Iframe invece di XMLHttpRequest per scambiare i dati con il server e, in altre implementazioni, tag <script> aggiunti dinamicamente (JSON);
- in genere viene usato XML come formato di scambio dei dati, anche se di fatto qualunque formato può essere utilizzato, incluso testo semplice, HTML preformattato, JSON e perfino EBML. Questi file sono solitamente generati dinamicamente da script lato server.

[ <https://it.wikipedia.org/wiki/AJAX> ]



<b><u>Estensione</u></b>	.xml
<b><u>Tipo MIME</u></b>	application/xml text/xml
<b><u>Sviluppato da</u></b>	<a href="#">W3C</a>
<b><u>Tipo</u></b>	<a href="#">Linguaggio di markup</a>
<b><u>Estensione di</u></b>	<a href="#">SGML</a>
<b><u>Esteso a</u></b>	<a href="#">XHTML</a> , <a href="#">RSS</a> , <a href="#">Atom</a> e altri
<b><u>Formato aperto?</u></b>	Sì

**XML** (sigla di **eXtensible Markup Language**, lett. "linguaggio di marcatura estendibile") è un [metalinguaggio](#) per la definizione di [linguaggi di markup](#), ovvero un linguaggio basato su un meccanismo sintattico che consente di definire e controllare il significato degli elementi contenuti in un documento o in un testo.

Il nome indica che si tratta di un linguaggio estendibile, in quanto permette di creare [tag](#) personalizzati e costituisce il tentativo di produrre una versione semplificata dello [Standard Generalized Markup Language](#) (SGML), che consente di definire nuovi linguaggi di markup.

[ <https://it.wikipedia.org/wiki/XML> ]



<b><u>Estensione</u></b>	.json
<b><u>Tipo MIME</u></b>	application/json
<b><u>Tipo</u></b>	<a href="#">Scambio dati</a>
<b><u>Estensione di</u></b>	<a href="#">JavaScript</a>
<b><u>Standard</u></b>	<a href="#">RFC 4627</a>
<b><u>Formato aperto?</u></b>	sì
<b><u>Sito web</u></b>	<a href="#">json.org</a>

**JSON** (pronuncia: 'dʒeɪ.sən), [acronimo](#) di **JavaScript Object Notation**, è un formato adatto all'interscambio di dati fra [applicazioni client/server](#).<sup>[1]</sup>

È basato sul linguaggio [JavaScript](#) Standard ECMA-262 3ª edizione (dicembre 1999), ma ne è indipendente. Viene usato in [AJAX](#) come alternativa a [XML/XSLT](#).<sup>[2]</sup>

[ [https://it.wikipedia.org/wiki/JavaScript\\_Object\\_Notation](https://it.wikipedia.org/wiki/JavaScript_Object_Notation) ]

# ATTORI LATO SERVER



Genere	<a href="#">Server web</a>	Tipo	<a href="#">no-profit</a>
<a href="#">Sviluppatore</a>	<a href="#">Apache Software Foundation</a>	Fondazione	giugno <a href="#">1999</a>
Data prima versione	1995	Fondatore	Brian Behlendorf, Ken Coar, Mark Cox, Lars Eilebrecht, Ralf S. Engelschall, Roy T. Fielding, Dean Gaudet, Ben Hyde, Jim Jagielski, Alexei Kosut, Martin Kraemer, Ben Laurie, Doug MacEachern, Aram Mirzadeh, Sameer Parekh, Cliff Skolnick, Marc Slomko, William Stoddard, Paul Sutton, Randy Terbush, Dirk-Willem van Gulik (attuale).
Ultima versione	2.4.46 (2020-08-07)	Sede centrale	<a href="#">Forest Hill (Maryland)</a>
<a href="#">Sistema operativo</a>	<a href="#">Multiplatforma</a>	Area di azione	Mondiale
<a href="#">Linguaggio</a>	<a href="#">C</a>		
<a href="#">Licenza</a>	<a href="#">Apache License (licenza libera)</a>		
<a href="#">Sito web</a>	<a href="#">httpd.apache.org</a>		

**Apache HTTP Server**, o più comunemente **Apache** ([IPA pronuncia](#): /a'pætʃi:/), è il nome di un [server web libero](#) sviluppato dalla [Apache Software Foundation](#). È la piattaforma server Web modulare più diffusa[1], in grado di operare su una grande varietà di [sistemi operativi](#), tra cui [UNIX/Linux](#), [Microsoft Windows](#) e [OpenVMS](#). È un [software](#) che realizza le funzioni di trasporto delle informazioni, di [internetwork](#) e di collegamento, ed ha il vantaggio di offrire funzioni di controllo per la [sicurezza](#) come quelle effettuate da un [proxy](#).

Operativamente, l'architettura è composta da un [demone](#) - in ambiente UNIX - o da un [servizio](#) - in ambiente Microsoft - il quale, sulla base delle impostazioni contenute nel file di configurazione *httpd.conf*, permette l'accesso a uno o più siti, gestendo varie caratteristiche di sicurezza e potendo ospitare diverse estensioni per pagine attive (o dinamiche), come [PHP](#) o [Jakarta/Tomcat](#).

Il [Web Server](#) Apache presenta un'architettura modulare, quindi ad ogni richiesta del [client](#) vengono svolte funzioni specifiche da ogni modulo di cui è composto, come unità indipendenti. Ciascun modulo si occupa di una funzionalità, ed il controllo è gestito dal core. Al di sopra del ciclo del core un [demone](#) esegue un ciclo di polling, attraverso il quale vengono interrogate continuamente le linee logiche da cui possono pervenire messaggi di richiesta. Il core passa poi la richiesta ai vari moduli in modo sequenziale, usando i parametri di uscita di un modulo come parametri di accesso per il successivo, creando così l'illusione di una comunicazione orizzontale fra i moduli ([Pipeline software](#)).

Le principali fasi di cui è composto il ciclo sono:

- *Translation*: traduce la richiesta del client
- *Access Control*: controlla le richieste in base ai criteri di autorizzazione

- *MIME Type*: identifica il tipo di contenuto e decide quali moduli possono contribuire a servire la richiesta
- *Response*: invia la risposta al client e attiva eventuali procedure
- *Logging*: tiene traccia di tutto ciò che è stato fatto

[ [https://it.wikipedia.org/wiki/Apache Software Foundation](https://it.wikipedia.org/wiki/Apache_Software_Foundation) ]  
 [ [https://it.wikipedia.org/wiki/Apache HTTP Server](https://it.wikipedia.org/wiki/Apache_HTTP_Server) ]



Data prima versione	1999 e 27 gennaio 2006	Tipo	<a href="#">no-profit</a>
Ultima versione	10.0.16 (20 gennaio 2022)	Fondazione	giugno <a href="#">1999</a>
Ultima <a href="#">alfa</a>	10.1.0-M10 (20 gennaio 2022)	Fondatore	Brian Behlendorf, Ken Coar, Mark Cox, Lars Eilebrecht, Ralf S. Engelschall, Roy T. Fielding, Dean Gaudet, Ben Hyde, Jim Jagielski, Alexei Kosut, Martin Kraemer, Ben Laurie, Doug MacEachern, Aram Mirzadeh, Sameer Parekh, Cliff Skolnick, Marc Slemko, William Stoddard, Paul Sutton, Randy Terbush, Dirk-Willem van Gulik (attuale).
<a href="#">Sistema operativo</a>	<a href="#">Multiplatforma</a>	Sede centrale	<a href="#">Forest Hill (Maryland)</a>
<a href="#">Linguaggio</a>	<a href="#">Java</a>	Area di azione	Mondiale
<a href="#">Licenza</a>	<a href="#">Licenza Apache 2.0</a> ( <a href="#">licenza libera</a> )		
<a href="#">Lingua</a>	<a href="#">Multilingua</a>		
<a href="#">Sito web</a>	<a href="http://tomcat.apache.org/">tomcat.apache.org/</a>		

**Apache Tomcat** (o semplicemente **Tomcat**) è un [server web](#) (nella forma di [contenitore servlet](#)) [open source](#) sviluppato dalla [Apache Software Foundation](#). Implementa le specifiche [JavaServer Pages](#) (JSP) e [servlet](#), fornendo quindi una [piattaforma software](#) per l'[esecuzione](#) di [applicazioni web](#) sviluppate in linguaggio [Java](#). La sua [distribuzione](#) standard include anche le funzionalità di web server tradizionale, che corrispondono al prodotto [Apache](#).

Tomcat non è un servizio che implementa completamente la specifica [Java EE](#), in quanto tale specifica, oltre le servlet ed alle JSP, supporta tantissime altre tecnologie. Quindi Tomcat non può essere considerato un [application server](#).<sup>[1]</sup> Anche se supporta solo parzialmente alcune tecnologie di [Java EE](#) (ovvero [Servlet](#) e [JavaServer Pages](#)), lo sviluppatore è libero di importarne altre come le [JPA](#) o altre tecnologie sempre in ambiente [Java EE](#). Tomcat può essere utilizzato anche come [contenitore servlet](#) per framework come [Spring framework](#).

[ [https://it.wikipedia.org/wiki/Apache Software Foundation](https://it.wikipedia.org/wiki/Apache_Software_Foundation) ]  
 [ [https://it.wikipedia.org/wiki/Apache Tomcat](https://it.wikipedia.org/wiki/Apache_Tomcat) ]





Genere	<a href="#">Framework per applicazioni web</a>	Tipo	<a href="#">no-profit</a>
Sviluppatore	<a href="#">Apache Software Foundation</a>	Fondazione	giugno <a href="#">1999</a>
Data prima versione	10 ottobre 2006 e 11 aprile 2014	Fondatore	Brian Behlendorf, Ken Coar, Mark Cox, Lars Eilebrecht, Ralf S. Engelschall, Roy T. Fielding, Dean Gaudet, Ben Hyde, Jim Jagielski, Alexei Kosut, Martin Kraemer, Ben Laurie, Doug MacEachern, Aram Mirzadeh, Sameer Parekh, Cliff Skolnick, Marc Slemko, William Stoddard, Paul Sutton, Randy Terbush, Dirk-Willem van Gulik (attuale).
Ultima versione	2.5.29 (22 gennaio 2022)	Sede centrale	<a href="#">Forest Hill (Maryland)</a>
Sistema operativo	<a href="#">Multiplatforma</a>	Area di azione	Mondiale
Linguaggio	<a href="#">Java</a>		
Licenza	<a href="#">Apache License 2.0</a>		
Sito web	<a href="#">struts.apache.org</a>		

**Apache Struts** è un [framework open source](#) per lo sviluppo di [applicazioni web](#) su [piattaforma Java EE](#). Inizialmente sviluppato come sotto-progetto di [Apache Jakarta](#), ma ora divenuto un progetto a sé, estende le Java [Servlet](#), incoraggiando gli sviluppatori all'utilizzo del [design pattern Model-View-Controller](#) (MVC). Il suo ideatore è [Craig McClanahan](#) ed è stato donato alla [Apache Software Foundation](#) nel maggio del 2000 da parte di [IBM](#).

L'utilizzo di Struts permette lo sviluppo di *web application* di notevoli dimensioni; inoltre agevola la suddivisione dello sviluppo del progetto fra vari sotto-team. In altre parole, i [web designer](#), e i vari gruppi di sviluppatori possono gestire in parallelo e autonomamente la loro parte del progetto. Tra le funzionalità offerte c'è la [i18N](#), ossia l'internazionalizzazione, una potente [tag library](#) e la validazione dei [form](#).

Per quanto riguarda il livello di modellazione dei dati sono supportati i [JavaBean](#) e [EJB](#). Struts lega i suoi componenti in base alle direttive contenute in un file XML, il file **struts-config.xml**. In questo file vengono configurati i componenti e le interazioni fra gli stessi.

[ [https://it.wikipedia.org/wiki/Apache\\_Software\\_Foundation](https://it.wikipedia.org/wiki/Apache_Software_Foundation) ]

[ [https://it.wikipedia.org/wiki/Apache\\_Struts](https://it.wikipedia.org/wiki/Apache_Struts) ]



# Pivotal

Genere	<a href="#">Integrated development environment</a>	Stato	<a href="#">Stati Uniti</a>
Sviluppatore	<a href="#">SpringSource</a>	Forma societaria	<a href="#">Public company</a>
Data prima versione	1° ottobre 2002	Borse valori	<a href="#">NYSE: PVTI</a>
Ultima versione	5.3.15 (13 gennaio 2022)	Fondazione	Aprile 2013
Sistema operativo	<a href="#">Multiplatforma</a>	Sede principale	<a href="#">San Francisco</a>
Linguaggio	<a href="#">Kotlin</a> <a href="#">Java</a> <a href="#">Groovy</a>	Gruppo	Dell Technologies
Licenza	<a href="#">Apache License 2.0</a> (licenza libera)	Settore	informatica
Sito web	<a href="https://spring.io/projects/spring-framework">spring.io/projects/spring-framework</a>	Prodotti	<ul style="list-style-type: none"><li><a href="#">software</a></li><li>servizi informatici</li></ul>
		Sito web	<a href="https://pivotal.io/">pivotal.io/</a>

**Spring** è un [framework open source](#) per lo sviluppo di [applicazioni](#) su [piattaforma Java](#).

Spring è stato largamente riconosciuto all'interno della comunità [Java](#) quale valida alternativa al modello basato su [Enterprise JavaBeans](#) (EJB). Rispetto a quest'ultimo, il framework Spring lascia una maggiore libertà al [programmatore](#) fornendo allo stesso tempo un'ampia e ben documentata gamma di soluzioni semplici adatte alle problematiche più comuni.

Sebbene le peculiarità basilari di Spring possano essere adottate in qualsiasi applicazione Java, esistono numerose estensioni per la costruzione di applicazioni *web-based* ([applicazioni web](#)) costruite sul modello della [piattaforma Java EE](#). Questo ha permesso a Spring di raccogliere numerosi consensi e di essere riconosciuto anche da importanti vendor commerciali quale framework di importanza strategica.

**Pivotal Software, Inc.** è una società di software e servizi con sede principale a [San Francisco](#), in [California](#), e più di venti altre sedi operative e regionali<sup>[1]</sup>. Le divisioni comprendono Pivotal Labs, che fornisce servizi di consulenza), Pivotal Cloud Foundry ed un gruppo che sviluppa prodotti di [big data](#).

[ [https://it.wikipedia.org/wiki/Spring\\_Framework](https://it.wikipedia.org/wiki/Spring_Framework) ]

[ [https://en.wikipedia.org/wiki/Spring\\_\(company\)](https://en.wikipedia.org/wiki/Spring_(company)) ]

[ <https://it.wikipedia.org/wiki/Pivotal> ]



Autore	PHP Group
Data di origine	8 giugno 1995
Ultima versione	8.1.3 (17 febbraio 2022)
Utilizzo	Applicazioni web lato server
<u>Paradigmi</u>	<u>Imperativo</u> , <u>funzionale</u> , <u>orientato agli oggetti</u> , <u>procedurale</u> , <u>riflessione</u>
<u>Tipizzazione</u>	Debole
<u>Estensioni comuni</u>	php
Influenzato da	<u>C++</u> , <u>Perl</u> , <u>C</u> , <u>Java</u> , <u>Tcl</u> , <u>HTML</u> e <u>JavaScript</u>
<u>Implementazione di riferimento</u>	
Implementazione	Interprete
<u>Sistema operativo</u>	<u>Multipiattaforma</u>
<u>Licenza</u>	<u>Licenza PHP</u>
<u>Sito web</u>	<u>www.php.net</u>

PHP (acronimo ricorsivo di "PHP: Hypertext Preprocessor", preprocessore di ipertesti; originariamente acronimo di "Personal Home Page"[1]) è un linguaggio di scripting interpretato, originariamente concepito per la programmazione di pagine web dinamiche. L'interprete PHP è un software libero distribuito sotto la PHP License[2].

Attualmente è principalmente utilizzato per sviluppare applicazioni web lato server, ma può essere usato anche per scrivere script a riga di comando o applicazioni stand-alone con interfaccia grafica. Un esempio di software scritto in PHP è MediaWiki, su cui si basano i progetti wiki della Wikimedia Foundation come Wikipedia.

Grazie soprattutto alla diffusione di Wordpress[3] (CMS scritto in PHP), secondo le statistiche, è il linguaggio lato server più usato al mondo[4][5][6].

[ <https://it.wikipedia.org/wiki/PHP> ]



<b>Genere</b>	<a href="#">Framework per applicazioni web</a>
<b>Sviluppatore</b>	<a href="#">Taylor Otwell</a>
<b>Data prima versione</b>	9 Giugno 2011
<b>Ultima versione</b>	9 (8 febbraio 2022)
<b>Sistema operativo</b>	<a href="#">Multiplatforma</a>
<b>Linguaggio</b>	<a href="#">PHP</a>
<b>Licenza</b>	<a href="#">licenza MIT</a> ( <a href="#">licenza libera</a> )
<b>Sito web</b>	<a href="#">laravel.com/</a>

**Laravel** è un [framework open source](#) di tipo [MVC](#) scritto in [PHP](#) per lo sviluppo di [applicazioni web](#), creato nel 2011 da [Taylor Otwell](#) come derivazione di [Symfony](#).

Distribuito con [licenza MIT](#), mantiene tutto il codice disponibile su [GitHub](#) e viene indicato, in base al punteggio GitHub e [StackOverflow](#), come il framework PHP più popolare<sup>[1]</sup>, seguito da [Symfony](#), [CodeIgniter](#) e altri; ad agosto 2014 risulta essere il progetto PHP più seguito su GitHub.<sup>[2]</sup>

Alcune delle caratteristiche sono: un [sistema di gestione dei pacchetti](#) modulare con un gestore delle dipendenze dedicato, differenti modalità di accesso ai [database relazionali](#), strumenti che aiutano la distribuzione e la manutenzione dell'applicazione, e la sua disposizione al [Syntactic sugar](#).

[ <https://it.wikipedia.org/wiki/Laravel> ]

## Sun Microsystems

**Sun Microsystems** (dal 2010 **Oracle America, Inc.**) è stata un'[azienda statunitense](#), con sede nella [Silicon Valley](#), produttrice di [software](#) e [semiconduttori](#) nota, tra le altre cose, per avere prodotto il [linguaggio di programmazione Java](#). Il 27 gennaio 2010 è stata acquistata dalla [Oracle Corporation](#) per 7,4 miliardi di dollari.

I prodotti Sun includevano [server](#) e [workstation](#) basati sulle [CPU SPARC](#), i [sistemi operativi SunOS](#) e [Solaris](#), il [file system](#) di rete [NFS](#), il [linguaggio di programmazione Java](#), la suite [OpenOffice.org](#) ed insieme ad [AT&T](#), Sun partecipò alla standardizzazione di [Unix System V Release 4](#). Prodotti di minor successo possono essere considerati il gestore di finestre [NeWS](#), la [GUI OpenLook](#) e le prime versioni di [Unix thin client](#) senza disco.

[ [https://it.wikipedia.org/wiki/Sun\\_Microsystems](https://it.wikipedia.org/wiki/Sun_Microsystems) ]



<b>Autore</b>	<a href="#">James Gosling</a> , <a href="#">Sun Microsystems</a>
<b>Data di origine</b>	<a href="#">23 maggio 1995</a>
<b>Ultima versione</b>	17 <a href="#">[1]</a>
<b>Utilizzo</b>	<a href="#">General purpose</a>
<b>Paradigmi</b>	<a href="#">Orientato agli oggetti</a>
<b>Tipizzazione</b>	<a href="#">Forte</a>
<b>Specifiche linguaggio</b>	di <a href="#">The Java Language Specification, Java SE 17</a>
<b>Estensioni comuni</b>	<a href="#">.jar</a> , <a href="#">.java</a> , <a href="#">.class</a>
<b>Influenzato da</b>	<a href="#">Simula</a> , <a href="#">Ada 83</a> , <a href="#">C++</a> , <a href="#">Eiffel</a> , <a href="#">Generic Java</a> , <a href="#">Mesa</a> , <a href="#">Modula-3</a> , <a href="#">Oberon</a> , <a href="#">Objective-C</a> , <a href="#">UCSD Pascal</a> , <a href="#">Smalltalk</a>
<b>Ha influenzato</b>	<a href="#">Ada 2005</a> , <a href="#">BeanShell</a> , <a href="#">C#</a> , <a href="#">Clojure</a> , <a href="#">D</a> , <a href="#">ECMAScript</a> , <a href="#">Groovy</a> , <a href="#">J#</a> , <a href="#">JavaScript</a> , <a href="#">PHP</a> , <a href="#">Python</a> , <a href="#">Scala</a> , <a href="#">Seed7</a> , <a href="#">Vala</a>
<b><a href="#">Implementazione di riferimento</a></b>	
<b>Implementazione</b>	interprete con JITter
<b>Sistema operativo</b>	<a href="#">Windows</a> , <a href="#">Linux</a> , <a href="#">macOS</a>
<b>Licenza</b>	<a href="#">GNU General Public License</a> , <a href="#">Java Community Process</a>

**Java** è un [linguaggio di programmazione ad alto livello](#), [orientato agli oggetti](#) e a [tipizzazione statica](#), che si appoggia sull'[omonima piattaforma](#) software di [esecuzione](#), specificamente progettato per essere il più possibile indipendente dalla [piattaforma hardware](#) di [esecuzione](#) (tramite [compilazione](#) in [bytecode](#) prima e [interpretazione](#) poi da parte di una [JVM](#)) (sebbene questa caratteristica comporti prestazioni in termini di computazione inferiori a quelle di linguaggi direttamente compilati come [C](#) e [C++](#) ovvero dunque perfettamente adattati alla piattaforma hardware).

## Librerie

### Versioni della piattaforma

Il linguaggio in sé definisce solo una minima parte delle [librerie](#) utilizzabili in combinazione con il linguaggio stesso. La parte restante è definita dalla piattaforma su cui il programma è eseguito. La Oracle mette a disposizione tre piattaforme ufficiali, ciascuna destinata ad ambiti diversi:

- [Java Platform, Standard Edition](#)
- [Java Platform, Micro Edition](#)
- [Java EE](#)

### Librerie di terze parti

In aggiunta il [programmatore](#) può utilizzare un numero arbitrario di librerie di terze parti. Queste librerie, contenute in vari [package](#), vengono utilizzate dal [programmatore](#) per utilizzare determinati [metodi](#) o [attributi](#) comuni per semplificare e uniformare i [programmi](#) e renderli maggiormente leggibili ai programmatori. Esistono numerosissimi package utilizzabili dai programmatori nel [linguaggio Java](#). Ad esempio esistono package:

- di [input e output](#) (*java.io*);
- per i [metodi](#) e le [costanti matematiche](#) (*java.Math*);
- per la creazione di applet (*java.applet*);
- per creare [applicazioni](#) di [rete](#) (*javax.net*);
- per la [stampa](#) (*javax.print*);
- per la [sicurezza](#) (*java.security* e *javax.security*).

[ [https://it.wikipedia.org/wiki/Java\\_\(linguaggio\\_di\\_programmazione\)](https://it.wikipedia.org/wiki/Java_(linguaggio_di_programmazione)) ]

## Java Platform, Standard Edition

**Java Platform, Standard Edition** è una [piattaforma software](#) ampiamente utilizzata nella [programmazione](#) in [linguaggio Java](#) per costruire e distribuire [applicazioni portatili](#) (di tipo *Desktop Application*) ad uso generale. Java SE consiste di una [macchina virtuale](#) ([Java Virtual Machine](#)), che deve essere usata per [eseguire programmi](#) Java, insieme ad una serie di [librerie](#) (o "pacchetti") o [API](#) necessari per consentire l'uso di [file system](#), [reti](#), [interfacce grafiche](#) e così via, all'interno di tali programmi.

La Java SE era conosciuta come *Java 2 Platform, Standard Edition* o J2SE dalla versione 1.2 fino alla versione 1.5. La sigla "SE" viene utilizzata per distinguere la piattaforma di base della versione Enterprise Edition di [Java EE](#) e [Java Micro Edition](#) ME. Il "2" è stato originariamente destinato a sottolineare i grandi cambiamenti introdotti nella versione 1.2, ma è stato rimosso nella versione 1.6. La convenzione di denominazione è stata cambiata più volte nel corso della storia delle versioni Java. A partire da J2SE 1.4 (Merlin).

La piattaforma *Java Enterprise Edition* (Java EE) è una specifica correlata che comprende tutte le classi in Java SE, oltre ad un numero di classi che sono più utili per i programmi che girano su [server](#) di postazioni di lavoro.

La piattaforma *Java Micro Edition* (Java ME) è una specifica correlata che ha lo scopo di fornire una raccolta certificata di [API](#) per lo sviluppo software di piccoli [dispositivi](#) con risorse limitate, come i [telefoni cellulari](#).

La [Java Development Kit](#) (JDK) e la [Java Runtime Environment](#) (JRE) sono i file effettivi che vengono [scaricati](#) e [installati](#) su un [computer](#) rispettivamente per sviluppare ed eseguire programmi Java.

La Java SE è stata organizzata in due parti fondamentali:

- Core Java
- Desktop Java

*Core Java* è la parte che contiene le API fondamentali del [linguaggio di programmazione](#), mentre *Desktop Java* è utilizzata per creare [GUI](#) di applicazioni o [Applet](#).

### Le librerie standard

Le librerie standard mirano ad includere le necessità più comuni per il [programmatore](#). Fra le più significative si possono citare

- la possibilità di costruire [GUI](#) (*interfacce grafiche*) con strumenti standard e non proprietari, utilizzando i package [JavaFX](#);
- la possibilità di creare applicazioni [multi-threaded](#), ovvero che svolgono in modo [concorrente](#) molteplici attività;
- il supporto per la [riflessione](#), ovvero la capacità di un programma di agire sulla propria struttura e di utilizzare classi caricate dinamicamente dall'esterno.

Fra gli argomenti che depongono spesso a favore di Java nella scelta del linguaggio di implementazione di un progetto software moderno, inoltre, si deve certamente contare la vastità delle librerie standard, che in particolare contribuiscono a renderlo altamente integrabile con altre tecnologie. Alcuni esempi delle funzionalità incluse tra le librerie standard di Java sono:

- accesso ai [database](#) tramite [JDBC](#) e ai [DBMS](#) con driver [ODBC](#) tramite il [bridge JDBC-ODBC](#);

- manipolazione documenti [XML](#);
- dialogo con piattaforme [CORBA](#);
- potenti strumenti per la programmazione [lato server](#) nel contesto [Web](#);
- supporto nativo per gran parte dei protocolli della famiglia [IP](#), vedi ad esempio il [Socket Java](#);
- supporto per le applicazioni [multimediali](#), [streaming](#) audio e video.

La vastità delle [librerie standard](#) è tale da renderne praticamente impossibile la padronanza "a memoria"; è dunque necessario avere a disposizione la documentazione delle [API](#), disponibile in linea sul sito ufficiale<sup>[1]</sup> o scaricabile sul computer sotto forma di archivio [compresso](#) (ad esempio, dalla pagina di archivio della [Sun Microsystem](#) dedicata a Java<sup>[2]</sup>).

## Versioni

Nel settembre 2017, Mark Reinhold, chief Architect della piattaforma Java, ha proposto di modificare il piano di rilascio da una release ogni due anni agli attuali sei mesi. Non tutte le versioni però hanno un supporto prolungato (LTS - Long Term Support), quest'ultime vengo rilasciate ogni 3 anni, attualmente sono LTS la versione 17, 11 e la 8.

Versione	Data rilascio	End Public [32]	of Updates	Free [29][30][31]	Extended Support Until
JDK Beta	1995	?			?
JDK 1.0	gennaio 1996	?			?
JDK 1.1	febbraio 1997	?			?
J2SE 1.2 Playground	dicembre 1998	?			?
J2SE 1.3 Kestrel	maggio 2000	?			?
J2SE 1.4 Merlin	febbraio 2002	ottobre 2008			febbraio 2013
J2SE 5.0 Tiger	settembre 2004	novembre 2009			aprile 2015
Java SE 6 Mustang	dicembre 2006	aprile 2013			dicembre 2018 dicembre 2026, supporto a pagamento per Azul Platform Core[33]
Java SE 7 Dolphin	luglio 2011	aprile 2015			luglio 2022
		gennaio 2019 per uso commerciale Oracle dicembre 2030 per uso non commerciale Oracle			
Java SE 8 (LTS)	marzo 2014	dicembre 2030 per Azul			dicembre 2030
		Almeno maggio 2026 per AdoptOpenJDK Almeno maggio 2026 per Amazon Corretto			
Java SE 9	settembre 2017	marzo 2018 per OpenJDK			Non disponibile
Java SE 10	marzo 2018	settembre 2018 per OpenJDK			Non disponibile
		settembre 2026 per Azul			
Java SE 11 (LTS)	settembre 2018	Almeno ottobre 2024 per AdoptOpenJDK			settembre 2026
		Almeno settembre 2027 per Amazon Corretto			settembre 2028 per Azul[33]
		Almeno ottobre 2024 per Microsoft[34][35]			
Java SE 12	marzo 2019	settembre 2019 per OpenJDK			Non disponibile
Java SE 13	settembre 2019	marzo 2020 per OpenJDK			Non disponibile
Java SE 14	marzo 2020	settembre 2020 per OpenJDK			Non disponibile
Java SE 15	settembre 2020	marzo 2021 per OpenJDK			N.D.
		marzo 2023 per Azul[33]			
Java SE 16	marzo 2021	settembre 2021 per OpenJDK			Non disponibile
Java SE 17 (LTS)	settembre 2021	settembre 2029 per Azul			settembre 2029 o successivo
		Almeno settembre 2027 per Microsoft			settembre 2031 per Azul Platform Prime
Java SE 18	marzo 2022	settembre 2022 per OpenJDK			Non disponibile



Java fu inizialmente distribuito come Java Development Kit 1.0 (JDK 1.0). Questo comprende il *runtime Java* (la virtual machine e le librerie di classi), e gli strumenti di sviluppo (il compilatore Java e altri strumenti). Successivamente, Sun fornì un pacchetto che comprendeva solo il runtime, chiamato Java Runtime Environment (JRE). Normalmente le persone si riferisc

ono ad una particolare versione di Java attraverso la versione del JDK (es. JDK 1.4) Le versioni JDK 1.2 e successive sono spesso chiamate Java 2. Per esempio, il nome ufficiale del JDK 1.4 è Java (TM) 2 Platform, Standard Edition 1.4.

Il linguaggio è rimasto stabile dal JDK 1.0 al JDK 1.4.x, con la [J2SE 5.0](#) sono state introdotte nuove funzionalità nel linguaggio. La libreria di classi che viene fornita con JDK è stata progressivamente ampliata e modificata in alcune parti.

### Versioni delle specifiche di linguaggio

I cambiamenti nel linguaggio sono formalizzati nelle specifiche di linguaggio. Ogni nuova edizione delle specifiche integra i cambiamenti richiesti fino a quel momento tramite le [Java Specification Request](#) e conseguentemente implementati nel compilatore javac.

- The Java Programming Language, First Edition
- The Java Programming Language, Second Edition
- The Java Programming Language, Third Edition
- The Java Programming Language, Java SE 7 Edition
- The Java Programming Language, Java SE 8 Edition
- The Java Programming Language, Java SE 9 Edition

[ [https://it.wikipedia.org/wiki/Java\\_Platform,\\_Standard\\_Edition](https://it.wikipedia.org/wiki/Java_Platform,_Standard_Edition) ]

## Jakarta EE

**Jakarta EE** (conosciuta, prima della versione 8, col nome di **Java EE**; e prima della versione 5, col nome di **Java 2 Enterprise Edition** o **J2EE**) è un insieme di specifiche le cui implementazioni vengono principalmente sviluppate in [linguaggio di programmazione Java](#) e ampiamente utilizzata nella [programmazione Web](#).

### Specifiche web

Le componenti che rientrano in questa categoria sono quelle legate alla visualizzazione delle pagine web. Nel caso in cui si vogliano usare solo le componenti lato web, non è necessario ricorrere ad un complesso referencing runtime di tipo full profile, ma è sufficiente usare un referencing runtime Web Profile o un più snello Web Server.

- [Servlet](#), ovvero la tecnologia che permette di trasmettere pagine [HTML](#) (ad esempio utilizzando la tecnologia [Java Server Pages](#)) al [client](#) e di esporre [web services](#);
- [Java Server Faces](#), un framework web basato su componenti e che permette lo sviluppo di siti con funzionalità AJAX.

### Specifiche enterprise

In questa categoria ci sono le componenti che sono specifiche di Jakarta EE. L'uso di queste componenti è richiesto solamente in applicativi di medie e grandi dimensioni.

- [Contexts and Dependency Injection](#) che permettono di usare l'inversione di controllo in modo molto simile a come si fa con [Spring framework](#) e forniscono la maggior parte delle funzionalità che viene fornita dagli Enterprise JavaBeans;
- Gli [Enterprise JavaBeans](#) definiscono un sistema a componenti distribuito che rappresentava il cuore della specifica Java EE. Tale sistema, infatti, forniva le tipiche caratteristiche richieste dalle applicazioni *enterprise*, come [scalabilità](#), sicurezza, [persistenza](#) dei dati e altro. Molte di tali caratteristiche sono oggi disponibili anche con Contexts and Dependency Injection;
- Il [Java Message Service](#) descrive un sistema per l'invio e la gestione di messaggi;
- [Java EE Security API](#) definisce la gestione degli utenti, delle password, dei gruppi e dell'autenticazione.

### Specifiche per i web service

In Jakarta EE è possibile realizzare sia [web service](#) di tipo [REST](#), che di tipo [SOAP](#). Le specifiche sono:

- [JSON Processing](#), per elaborare informazioni codificate in formato [JavaScript Object Notation](#);
- [JSON Binding](#), per trasformare oggetti Java in stringhe [JavaScript Object Notation](#) e vice versa;
- [RESTful Web Services](#), per realizzare web services mediante una architettura di tipo [Representational State Transfer](#)
- [JAX-WS](#) per i web service di tipo [SOAP](#)
- [Java Architecture for XML Binding](#) per trasformare oggetti Java in stringhe [XML](#) e vice versa.

### Specifiche per l'interazione con la base dati

Questa categoria contiene le tecnologie che permettono la persistenza dei dati all'interno di una base

dati relazionale. Anche le componenti di questa categoria possono essere usate al di fuori di un referencing runtime.

- Il [Java Transaction API](#) è un sistema per il supporto delle [transazioni distribuite](#);
- Il [Java DataBase Connectivity](#) è un'interfaccia per l'accesso a qualsiasi tipo di [basi di dati](#) (compresa anche in [Java Platform, Standard Edition](#));
- Il [Java Persistence API](#) è l'insieme di classi che possono essere usate per la gestione della persistenza dei dati.

[ [https://it.wikipedia.org/wiki/Jakarta\\_EE](https://it.wikipedia.org/wiki/Jakarta_EE) ]

## JSP

<u>Estensione</u>	.jsp
<u>Tipo MIME</u>	application/jsp
<u>Sito web</u>	<a href="http://www.oracle.com/technetwork/java/jsp-138432.html">www.oracle.com/technetwork/java/jsp-138432.html</a>

**Java Server Pages**, di solito indicato con la sigla **JSP** (letto anche talvolta come **Java Scripting Preprocessor**), è una tecnologia di [programmazione web](#) in [Java](#) per lo sviluppo della [logica di presentazione](#) (tipicamente secondo il pattern [MVC](#)) di [applicazioni web](#), fornendo [contenuti dinamici](#) in formato [HTML](#) o [XML](#). Si basa su un insieme di speciali [tag](#), all'interno di una pagina HTML, con cui possono essere invocate [funzioni](#) predefinite sotto forma di [codice Java](#) ([JSTL](#)) e/o funzioni [JavaScript](#). In aggiunta, permette di creare librerie di nuovi tag che estendono l'insieme dei tag standard (JSP [Custom Tag Library](#)). Le librerie di tag JSP si possono considerare estensioni [indipendenti dalla piattaforma](#) delle funzionalità di un [server web](#).

Nel contesto della piattaforma Java, la tecnologia JSP è correlata con quella delle [servlet](#): all'atto della prima invocazione, le pagine JSP vengono infatti [tradotte](#) automaticamente da un [compilatore JSP](#) in [servlet](#). Una pagina JSP può quindi essere vista come una rappresentazione ad alto livello di un servlet. Per via di questa dipendenza concettuale, anche l'uso della tecnologia JSP richiede la presenza, sul [server web](#), di un [servlet container](#), oltre che di un server specifico JSP detto [motore JSP](#) (che include il compilatore JSP); in genere, servlet container e motore JSP sono integrati in un unico prodotto (per esempio, [Tomcat](#) svolge entrambe le funzioni).

JSP è una tecnologia alternativa rispetto a numerosi altri approcci alla generazione di pagine web dinamiche, per esempio [PHP](#), o [ASP](#) o la più tradizionale [CGI](#). Differisce da queste tecnologie non tanto per il tipo di contenuti dinamici che si possono produrre, quanto per l'architettura interna del software che costituisce l'applicazione web (e, di conseguenza, sui tempi di sviluppo, la portabilità, la [modificabilità](#), le prestazioni, e altri aspetti di [qualità del software](#)).

[Sun Microsystems](#) raccomanda di utilizzare il [pattern Model-View-Controller](#) con le pagine JSP in modo da dividere il livello di presentazione da quello dell'elaborazione della request e dalla memorizzazione dei dati. Le normali [servlet](#) o delle pagine JSP dedicate vengono utilizzate per processare i dati. Dopo che l'elaborazione è terminata, il controllo passa ad una pagina JSP che serve solo a visualizzare l'output. Quest'ultima pagina JSP dovrebbe contenere solo [HTML](#), [XML](#) e action e tag JSP; la pagina dovrebbe far uso dei [JavaBeans](#) per ottenere i dati.

In altri termini nello sviluppo di un'[applicazione web](#) la convenzione vuole che nelle JSP ci sia meno codice Java possibile e quello presente vada a richiamare codice Java nativo (oggetti e [metodi](#)) implementato in classi separate apposite dette appunto JavaBeans. Questa separazione consente infatti un facile [riuso di codice](#) dei Java beans una volta richiamato in un qualsiasi punto richiesto dell'applicazione web.

Una **JSP** è un documento di testo, scritto con una [sintassi](#) specifica, che rappresenta una [pagina web](#) di contenuto parzialmente o totalmente dinamico. Elaborando la pagina JSP, il motore JSP produce dinamicamente la pagina HTML finale che verrà presentata al [web browser](#) dell'utente. La pagina JSP può contenere tre tipi di elementi, a cui corrispondono tre diversi modi di elaborazione: *contenuti statici*, *direttive* e *script*.

[ [https://it.wikipedia.org/wiki/JavaServer\\_Pages](https://it.wikipedia.org/wiki/JavaServer_Pages) ]

## Servlet

I **servlet** sono oggetti scritti in linguaggio Java che operano all'interno di un server web (es. Tomcat, Jetty) oppure un server per applicazioni (es. WildFly, GlassFish) permettendo la creazione di applicazioni web[1] (elaborazione lato server). Il nome deriva in contrapposizione alle Java applet, piccoli programmi scritti in Java che si eseguono all'interno del browser dell'utente client (elaborazione lato client).

L'uso più frequente delle servlet è la generazione di pagine web dinamiche a seconda dei parametri di richiesta inviati dal client browser dell'utente al server. Nelle applicazioni più moderne non viene eseguita la programmazione diretta delle servlet, ma si preferisce usare dei framework web che implementano la specifica servlet, oppure delle JavaServer Pages che vengono poi tradotte (compilate) in servlet a runtime.

I programmi che implementano le specifiche dei servlet possono girare all'interno di qualunque *servlet container* e non sono vincolati ad un particolare server. Lo standard delle servlet rientra all'interno di un vasto insieme di standard detto Java EE[2][3]. Una servlet può avere molteplici funzionalità e può essere associata ad una o più risorse web.

[ <https://it.wikipedia.org/wiki/Servlet> ]

## Model-View-Controller

**Model-View-Controller (MVC)**, talvolta tradotto in [italiano](#) con la dicitura **modello-vista-controllo**), in [informatica](#), è un [pattern architetturale](#) molto diffuso nello sviluppo di sistemi [software](#), in particolare nell'ambito della [programmazione orientata agli oggetti](#) e in [applicazioni web](#), in grado di separare la [logica di presentazione](#) dei dati dalla [logica di business](#).<sup>[1]</sup> Questo pattern si posiziona nel livello logico o di business e di presentazione in una [architettura multi-tier](#).

Il pattern è basato sulla separazione dei compiti fra i componenti [software](#) che interpretano tre ruoli principali:

- il *model* fornisce i [metodi](#) per accedere ai dati utili all'applicazione;
- la *view* visualizza i dati contenuti nel model e si occupa dell'interazione con utenti e agenti;
- il *controller* riceve i comandi dell'utente (in genere attraverso la view) e li attua modificando lo stato degli altri due componenti.

[ <https://it.wikipedia.org/wiki/Model-view-controller> ]

## RANKING

[ <https://trends.google.it/trends/?geo=IT> ]

[ <https://www.similarweb.com> ]

## TIOBE Programming Community Index

Il TIOBE Programming Community Index, in breve TPC Index, è un indicatore della popolarità dei linguaggi di programmazione, la classifica è compilata grazie ai dati ricavati dai tre motori di ricerca [Google](#), [MSN](#) e [Yahoo!](#) valutando la disponibilità mondiale di esperti programmatori, corsi a tema e i fornitori terze parti.

TIOBE, acronimo di The Importance Of Being Earnest (di [Oscar Wilde](#)), è una compagnia olandese fondata nell'anno [2000](#) che si annuncia come "The coding standard company" e si propone in qualità di consulente per il rispetto degli standard nella programmazione. Gli organizzatori del TPC Index fanno sapere che la loro classifica non può e non vuole indicare il linguaggio di programmazione migliore e neppure il linguaggio con il quale sono scritte il maggior numero di linee di codice.

[ [https://it.wikipedia.org/wiki/TIOBE\\_Programming\\_Community\\_Index](https://it.wikipedia.org/wiki/TIOBE_Programming_Community_Index) ]

[ <https://www.tiobe.com/tiobe-index/> ]

## Common Vulnerabilities and Exposures

Il **Common Vulnerabilities and Exposures**, o **CVE** (in italiano **Vulnerabilità ed esposizioni comuni**), è un dizionario di vulnerabilità e falle di sicurezza note pubblicamente. È mantenuto dalla [MITRE Corporation](#) ed è finanziato dalla [National Cybersecurity FFRDC](#) del [Dipartimento della Sicurezza interna degli Stati Uniti](#).<sup>[1]</sup> Il CVE è utilizzato dal [Security Content Automation Protocol](#) (SCAP) e le vulnerabilità, identificate da un identificatore univoco, sono elencate nel sistema MITRE e nel [National Vulnerability Database](#) americano. L'identificazione univoca delle CVE permette una maggiore comunicazione nel mondo della sicurezza e aiuta nella valutazione della diffusione di servizi e strumenti.<sup>[2]</sup>

Come detto nel sito CVE, il principale metodo per richiedere l'inserimento di una vulnerabilità tra quelle conosciute è farne richiesta a una CNA (CVE Numbering Authorities), organizzazioni di ricerca, ricercatori nel campo della sicurezza e produttori di sistemi operativi. Tra essi possiamo trovare: [Adobe](#), [Apple](#), [Attachmate](#), [BlackBerry](#), [CERT Coordination Center](#), [Cisco](#), [Debian GNU/Linux](#), [Distributed Weakness Filing Project](#), [EMC](#), [FreeBSD](#), [Google](#), [HP](#), [IBM](#), [ICS-CERT](#), [JPCERT/CC](#), [Juniper](#), [Microsoft](#), MITRE (CNA primaria), [Mozilla](#), [Oracle](#), [Red Hat](#), [Silicon Graphics](#), [Symantec](#) e [Ubuntu](#).<sup>[3]</sup>

[ [https://it.wikipedia.org/wiki/Common\\_Vulnerabilities\\_and\\_Exposures](https://it.wikipedia.org/wiki/Common_Vulnerabilities_and_Exposures) ]

[ <https://www.mitre.org/> ]

[ <https://nvd.nist.gov/vuln/search> ]