

Flask vs. Django: Die beiden Python-Frameworks im Vergleich

IONOS Redaktion

20.09.2022

9 mins



:≡ Inhaltsverzeichnis

Django und [Flask](#) zählen zu den beliebtesten [Webframeworks](#) für Python. Während Flask sehr leicht aufgebaut ist, stellt Django viele Funktionen zur Verfügung.

Website mit eigener Domain – MyWebsite Now

Erstellen Sie Ihre eigene Website in wenigen Minuten

Professionelle Templates

Änderungen am Design mit nur einem Klick

Zu den Tarifen

Domain, SSL und E-Mail-Postfach inklusive

Was ist Python?

Python ist eine universelle Programmiersprache, die erstmals 1991 veröffentlicht wurde. Sie verfolgt das klare Ziel, kürzeren und dadurch **besser lesbaren Code** zu verwenden. Die dynamische Sprache wird heute von der gemeinnützigen Python Software Foundation (PSF) verwaltet und mithilfe einer großen Community stetig

Digital Guide

Tipp

Sie sind bereits mit Python vertraut und möchten sich eingehender mit der Programmiersprache beschäftigen? In diesem Fall können Sie z. B. [Python für Web Scraping nutzen](#) oder [Python als CGI-Script ausführen](#).

Welche Webframeworks gibt es für Python?

Neben Django und Flask existiert eine große Anzahl unterschiedlicher Python-Webframeworks. Diese unterscheiden sich durch ihre Zielsetzung, ihren Aufbau und/oder ihr primäres Aufgabengebiet. So gibt es **Full-Stack-Frameworks**, die bereits komplett Werkzeugkästen für Apps und andere Software enthalten, dadurch aber auch deutlich umfangreicher ausfallen. Prominente Beispiele für diese Frameworks sind TurboGears und Giotto.

Andere Anbieter setzen auf **Non-Full-Stack-Lösungen**. Sie stellen nur die Basics zur Verfügung und erlauben Nutzerinnen und Nutzern so die maximale Freiheit bei der Erstellung neuer Anwendungen. Gleichzeitig ist dadurch aber auch deutlich mehr Eigenleistung nötig. Bekannte Beispiele sind CherryPy oder Growler.

Tipp

In nur drei Schritten zur perfekten Website: Mit [MyWebsite](#) von IONOS bekommen Sie einen praktischen Baukasten, mit dem Sie innerhalb kürzester Zeit Ihre eigene professionelle Homepage kreieren und veröffentlichen. Zahlreiche nützliche Features sind dabei im Preis enthalten.

Digital Guide

Sowohl Django als auch Flask sind kostenlos, [Open Source](#) und verfügen weltweit über eine begeisterte Anhängerschaft.

Django wurde 2005 entwickelt und ist ein plattformübergreifendes **Full-Stack-Framework mit BSD-Lizenz**. Ursprünglich als reines CMS erdacht – es gibt dafür heute [django CMS](#) –, eignet es sich mittlerweile für eine Vielzahl unterschiedlicher Web-Applikationen. Django verfolgt das DRY-Prinzip („Don't repeat yourself“) und bietet eine **Rundum-Lösung** für den Aufbau und Unterhalt von Websites oder anderen Webentwicklungen. Als verlässliches und gut dokumentiertes Framework wird Django u. a. von Instagram, Spotify und YouTube genutzt.

Im Gegensatz zu Django verfolgt das Micro-Framework Flask des österreichischen Entwicklers Armin Ronacher seit seiner Erstveröffentlichung 2010 einen ausdrücklich **minimalistischen Ansatz**. Einzig die Bibliothek „Werkzeug“ zur Erstellung von WSGI-Anwendungen sowie die Template-Engine Jinja sind enthalten. Weitere Komponenten stellt Flask nicht zur Verfügung. Entwicklerinnen und Entwickler können allerdings bestehende Bibliotheken einfach integrieren. So werden **zahlreiche Funktionen** zur Verfügung gestellt, wobei der Kern von Flask trotzdem schlank bleibt. Auch Flask verfügt über eine freie BSD-Lizenz und wird u. a. von Netflix, Reddit und Airbnb verwendet.

Tipp

Mit [MyWebsite Now](#) von IONOS erstellen Sie in nur wenigen Minuten Ihre professionelle Homepage. Dafür müssen Sie nur ein Design wählen, Inhalte hinzufügen und Ihren Onlineauftritt veröffentlichen. Domain, SSL und E-Mail-Postfach sind inklusive.

Flask vs. Django: Der direkte Vergleich

Digital Guide

große Unterschied zwischen den beiden Frameworks. Während Django nämlich über zahlreiche Features verfügt, ist das bei Flask zunächst nicht der Fall. Letzteres kann allerdings die meisten benötigten Funktionen extern zur Verfügung stellen. Django verfügt über ein **Standard-ORM** und unterstützt z. B. [MySQL](#) und [PostgreSQL](#) direkt. Das Framework kann für nahezu alle Webformate verwendet werden und funktioniert auch in Kombination mit anderen Frameworks auf Client-Seite. Flask verzichtet auf einen ORM und erlaubt stattdessen den Zugriff auf die **Features anderer Bibliotheken**.

Sicherheit

Django und Flask gelten beide aus unterschiedlichen Gründen als **sehr sichere Lösungen**. So ist Django mit zahlreichen Sicherheitsfeatures wie CSRF, SQL und XSS ausgestattet und verfügt darüber hinaus über ein engagiertes Team, das eventuelle Lecks und Bugs frühzeitig identifiziert und behebt. Viele weitere sicherheitsrelevante Funktionen werden dazu automatisch zur Verfügung gestellt und durchgeführt.

Das ist bei Flask naturgemäß nicht der Fall. Da der Code allerdings deutlich kompakter ist, gibt es an der Basis auch weniger Sicherheitsrisiken. Diese entstehen eher bei den zahlreichen Erweiterungen von Dritten. Flask ist nur wirklich sicher, wenn Sie alle **notwendigen Sicherheitsupdates** im Blick behalten und dafür sorgen, dass sich sämtliche Bibliotheken immer auf dem neuesten Stand befinden. Anders als Django übernimmt Flask diese Aufgaben nicht für Sie, sodass die Sicherheit Ihrer Anwendung und die Vermeidung klassischer [Python-Probleme](#) im Vergleich deutlich mehr Aufwand erfordern.

Flexibilität

Mehr Flexibilität als bei Flask ist kaum möglich. Das Webframework stellt Ihnen lediglich eine Basis zur Verfügung und erlaubt es Ihnen dann, jene Bibliotheken zu verwenden, die Sie benötigen. Mit dem notwendigen Vorwissen haben Sie so die Möglichkeit, jedes neue Projekt ganz individuell aufzubauen. Plugins und zusätzliche Funktionen können ebenfalls **nach Belieben eingesetzt werden**. Das ist zwar verlockend, erfordert allerdings auch mehr Arbeit und

Digital Guide

weniger flexibel, dafür aber ein wenig benutzerfreundlicher.

Performance

Auch beim Blick auf die Performance muss man die beiden unterschiedlichen Ansätze berücksichtigen. Im oberflächlichen Vergleich ist Flask Django hier leicht voraus. Das ist in erster Linie der jeweiligen Größe geschuldet. Als Leichtgewicht ist **Flask ein wenig schneller**, während Django zumindest in manchen Fällen ein bisschen schwerfälliger daherkommt. Das kann sich allerdings schnell ändern, wenn Sie mehr externe Funktionen und Plug-ins hinzufügen. Diese können die Performance unter Umständen schmälern und dafür sorgen, dass sich die beiden Frameworks noch mehr angleichen.

Lernkurve

Erste Erfolge erzielen Sie schneller mit Flask. Das Framework ist deutlich kleiner und daher leichter zu bedienen. Bei Django benötigen Sie hingegen eine **längere Eingewöhnungszeit**, da es dauert, bis man die Struktur und Besonderheiten verinnerlicht hat. Im weiteren Verlauf kann sich dieses Bild allerdings ins Gegenteil verkehren. Haben Sie sich einmal in Django eingearbeitet, werden Sie merken, dass **viele Abläufe automatisch funktionieren**. Im Vergleich Flask vs. Django kann es dann sein, dass die Verwendung von Plug-ins beim leichteren Kontrahenten durchaus einige Fallstricke mit sich bringt. Möchten Sie beide Frameworks ausprobieren, sollten Sie mit Flask anfangen und dann zu Django übergehen.

Community

Sei es zu Beginn Ihrer Arbeit mit einem Webframework oder nach den ersten erfolgreichen Projekten: Wenn Sie Fragen haben und Hilfe benötigen, sind eine **gute Dokumentation und eine aktive Community** sehr wichtig. Nur so landen Sie nicht in einer Sackgasse, sondern finden eine Lösung für Ihr jeweiliges Problem. In dieser Hinsicht sind Django und Flask beide vorbildlich. Sie werden für viele Problemstellungen bereits **hilfreiche User-Cases** finden und sollten Sie doch einmal Support benötigen, steht Ihnen bei beiden Frameworks eine große Community mit Rat und Tat zur Seite.

Digital Guide

Einsatzmöglichkeiten und ihrer überzeugenden Sicherheitsarchitektur liegt. Die oben bereits genannten Beispiele zeigen, dass auch große Unternehmen auf die Qualitäten von Django und Flask setzen.

💡 Tipp

Entscheiden Sie sich für [Webhosting](#) von IONOS und profitieren Sie von zahlreichen nützlichen Features sowie flexibler Skalierbarkeit. SSL-Zertifikat und ein großer Speicher sind dabei immer enthalten.

Django und Flask auf einen Blick

	Flask	Django
Umfang	Micro-Framework mit vielen Anpassungsoptionen	Macro-Framework mit vielen standardmäßigen Features
Sicherheit	Sicher, allerdings abhängig von manueller Pflege	Sicher
Flexibilität	Sehr flexibel	Weniger flexibel
Performance	Im Vergleich etwas besser	Gute Performance, aber etwas langsamer
Lernkurve	Leichter Einstieg	Großer Umgang, daher schwieriger Einstieg
Community	Große Community	Große Community
Verbreitung	Sehr beliebt	Sehr beliebt

Digital Guide

Um für Ihre Zwecke die beste Wahl zu treffen, ist es wichtig, sich die jeweiligen Vor- und Nachteile von Flask vs. Django anzusehen:

Flask

Vorteile	Nachteile
✓ Sehr schlank	✗ Es gibt weniger Hilfestellung
✓ Es stellen sich schnelle Lernerfolge ein	✗ Der Aufwand für die Pflege ist größer
✓ Flexibilität ist unvergleichlich	✗ Die Abhängigkeit von Drittanbietern ist groß
✓ Framework ist sehr schnell	

Django

Vorteile	Nachteile
✓ Viele Vorgaben und Funktionen	✗ Weniger flexibel
✓ Sehr sicheres Framework	✗ Teilweise nicht kompatibel mit bestimmten Plugins
✓ Kompatibel mit zahlreichen anderen Frameworks	✗ Schwerfälliger
✓ Umfangreiche Admin-Tools	✗ Kompliziertere Lernkurve
✓ Standard-ORM	

Wann sollte man welches Framework verwenden?

Digital Guide

umfangreiches Projekt starten und dabei möglicherweise sogar mit anderen Entwicklerinnen und Entwicklern zusammenarbeiten. Es ist eine sichere und sehr funktionale Lösung. Für kleinere Projekte, erste Gehversuche und Seiten mit wechselnden Ansprüchen ist hingegen Flask die bessere Wahl. Seine Flexibilität macht das Framework zu einer **starken Alternative für Einsteiger und Profis** gleichermaßen.

Flask vs. Django: Fazit

Es lässt sich nicht klar feststellen, welches Framework das bessere ist, da die Ansprüche von Django und Flask für einen echten Vergleich viel zu unterschiedlich sind. Wenn Sie sich daher bei der Frage [PHP vs. Python](#) für Letzteres entschieden haben, lohnt es sich, beide Frameworks kennenzulernen und dann von Projekt zu Projekt zu entscheiden. So finden Sie immer die beste Lösung.

💡 Tipp

In unserem Digital Guide finden Sie alles Wichtige rund um die Themen Webentwicklung und Webdesign: Erfahren Sie, was die neuesten [Webdesign-Trends](#) sind, entdecken Sie die nützlichsten [Web Components](#) und lernen Sie alles über [User-Experience-Design](#). Falls Sie sich speziell für die Programmiersprache CSS interessieren, empfehlen wir Ihnen folgende Artikel:

- [CSS lernen](#)
- [CSS-Tricks](#)
- [CSS in HTML](#)
- [CSS Media Queries](#)

War dieser Artikel hilfreich?



Digital Guide



KI-Wissen für Ihren Erfolg

Unser Newsletter bringt Ihnen die wichtigsten Insights rund um KI – verständlich, praxisnah und auf den Punkt.

[Jetzt anmelden](#)

Beliebte Artikel

Was ist eine E-Mail Domain und wie wird diese eingerichtet?

Setzen Sie auf professionelle E-Mail-Kommunikation! Wir zeigen Ihnen die Vorteile einer...

Digital Guide

WIE KAUT MAN EINE DOMAIN:

Eine Anleitung

Wie registriert und sichert
man sich eine Domain mit
gewünschter Top- und...

Mehr lesen →

Welche Domaintypen gibt es?

Welche Domainendungen
gibt es? Und worin liegt der
Unterschied zwischen Top-
Level-Domain...

Mehr lesen →

Prompt Engineering erklärt

Was ist Prompt Engineering,
wie lassen sich damit die
Ergebnisse von ChatGPT und
Co....

Mehr lesen →

7 Website-Typen im Überblick: Welche Website brauchen Sie?

Den richtigen Website-Typen
zu wählen, ist entscheidend
für den Erfolg jedes...

Mehr lesen →

Ähnliche Artikel

Digital Guide



Alchemy CMS

Während die meisten Content-Management-Systeme auf der Programmiersprache PHP basieren, nutzt Alchemy CMS die



CMS ohne Datenbank: die simple Lösung für kleine Webprojekte

Kennen Sie Content Management Systeme ohne Datenbank? Diese einfachen CMS-Systeme sind in der Regel kostenlos, können online heruntergeladen und oftmals nach Belieben...

Datenbank

Mehr lesen →

Digital Guide

[Cookies](#)

[Über IONOS](#)

[Jobs](#)

[Newsroom](#)

[Hilfe-Center](#)

[Startup Guide](#)

[AGB](#)

[Datenschutz](#)

[Impressum](#)

[Digital an Ihrer Seite](#)

© 2025 IONOS SE