The Python programming language comes with a wide assortment of web frameworks that can be used by web developers to build websites. This allows the web developer to choose the framework that most closely fits their task and is most suitable for the endeavour. Among many popular choices, Django and Flask are the two that are most often compared in order of popularity. Most likely, this is the result of both of them sharing some similarities and many differences. Each framework has its own unique features, so we can use it in accordance with the requirements of a particular project. As a full-stack web framework, Django is best suited for developing large and complex web applications, while Flask is a lightweight, extensible framework that allows you to develop small web applications. With Django, you will enjoy the batteries-included approach and have access to the most comprehensive functionality.

Are you still unsure which framework to use for web development? Even though each of these web development frameworks has its own unique features, there are many factors that you should consider before choosing one for your application.

What is Django?

Created by Adrian Holovaty and Simon Willison in the year 2003, Django is a Python-based open-source framework for designing web applications. It is a high-level web framework that is built to make the web development process faster and more efficient. Inspired by many of the old frameworks like CherryPy, Zope, Plone, etc. Django is a free source with enhanced features and better performance. Developers choose Django because it enables them to use it for standard functionalities with a limited interference of systems, protocols, and management.

Confused about your next job?

Expand in New Tab

Django is also called a 'framework for fussbudgets with deadlines' as its framework encourages rapid development and clean, pragmatic design. The agile development process of the framework aims solely to provide quality with rapidness and efficiency. Django deals with some of the basic development functions quickly like site maps, content organization, client information, and, so many more. It just focuses on finishing the application as quickly as possible.

Companies Using Django
Django is used by the following giant companies:

- Instagram
- Coursera
- Mozilla
- Pinterest
- National Geographic
- Spotify
- Udemy
- Zapier, etc.

Key Features: Django

Some of the key features of Django are as follows:

- **Fast**: It is insanely Fast. Without any thought, the Django working process from concept to completion is extremely fast.
- **Versatile**: Django is a versatile framework that enables developers to work on different platforms varying from content management systems like WordPress, etc., to social network sites like LinkedIn, Youtube, etc., to news sites like The New York Times, CNN, etc.
- **Adaptable:** Django is adaptable to different formats like JSON, HTML, XML, and many more.
- **Scalable:** It is a framework that ensures scalability (a system that allows making changes in different layers and updations without much cost and effort i.e., every layer is independent) and maintenance (the design and code are not susceptible to duplications and, hence, the code can be reused and maintained properly)Secure: Django guarantees security with powerful authentication systems and protocols to avoid clickjacking, unauthorized access, cyberattacks, etc.
- **Portable:** Django is a Python-based framework and, therefore,

portable.

What is Flask?

Flask is also a Python-based microframework that is used for web application development. It was introduced by Armin Ronacher in the year 2011 as a trial method of joining two solutions i.e., Werkzeug (a server framework) and Jinja2 (a template library).

It was supposed to be a trial run in a zip file that ultimately originates from the positive influence of Flask.

Flask is categorized as a micro framework because it does not depend on external libraries to perform the tasks of a framework. It has its tools, technologies, and libraries to support the functionalities of web application development.

Since this framework is more independent and flexible, many developers prefer to start with Flask.

Companies using Flask
Flask is used by the following giant companies:

- Netflix
- Airbnb
- MIT
- Reddit
- Lyft
- Zillow
- Mozilla
- MailGui, etc.

Key Features: Flask

Some of the features of Flask are:

- **Lightweight**: It is a lightweight framework as it is independent of external libraries. It gives a quick start to the web development process of complex applications.
- **Independent**: Flask gives independent or full control to the developer for creating applications. You can experiment with the architecture or the libraries of the framework.
- **Integrated Unit Testing:** Flask's integrated unit testing system enables faster debugging, robust development, and freedom to experiment.
- **Secure Cookies:** A secure cookie is an attribute of an HTTP request that enables the security of channels and ensures no unauthorized person has access to the text. Flask supports the feature of secure cookies.
- **Compatible:** Flask is compatible with the latest technologies like Machine Learning, Cloud, etc.
- **Flexible and Scalable:** Support WSGI templates that allow flexibility and scalability for web applications.
- It comes with a built-in server and debugger.
- Simple and adaptable configurations

Python Flask vs Django: Tabular Difference

After reading in detail about both Python-based frameworks,

Django and Flask, you must have understood that there are as many similarities as differences.

Therefore, for better judgment and deciding upon which framework is the best one, you need to look at the head-to-head comparison of the frameworks that highlight the difference between Flask and Django.

Find below the difference between Django vs Flask

Parameter	Django	Flask
Type of Framework	Django is a full-stack web framework that enables ready-to-use solutions with its batteries-included approach.	Flask is a lightweight framework that gives abundant features without external libraries and minimalist features.
Working of Framework/Data Model	Django follows an object-oriented approach that enables object-relational mapping (linking databases and tables with classes)	Flask works on a modular approach that enables working through outsourced libraries and extensions.
Project Layout	Django is suitable for multiple-page applications.	Flask is suitable for only single-page applications.
Bootstrapping	-Django-admin is the built-in bootstrapping tool of Django that	Flask does not come

Tool	allows the creation of web applications without any external input.	with an in-built bootstrapping tool.
Database Support	Django supports the most popular relational database management systems like MySQL, Oracle etc.	Flask does not support the basic database management system and uses SQLAlchemy for database requirements.
Flexibility	Django is less flexible because of its in-built features and tools. Developers cannot make changes to the modules.	Flask is a micro-based framework with extensible libraries making itself a flexible framework for developers.
Template Engine	Django is inspired by the Ninja2 template but has its built-in model view template that makes the development process easier.	Flask used Ninja2 template design
Control	Developers do not have full control over the modules and functions of Django because of built-in libraries.	Flask allows developers full control over the creation of applications with no dependencies from external libraries.

Working Style	The working style of Django is Monolithic	The working style of Flask is diversified style.
Debugger	Django does not support any virtual debugging.	Flash has an in-built debugger that offers virtual debugging
Routing and Views	The Django framework supports the mapping of URLs to views through a request.	Flask web framework allows mapping of URL to class-based view with Werkzeug.
Structure	The Django framework structure is more conventional.	The Flask web framework structure is random.
HTML	Django supports dynamic HTML pages	The Flask framework does not support dynamic HTML pages
Best Features	• Open-Source Great Community • Fast Development • Easy to learn • Secure	• Extensive Documentation • Lightweight • Minimal Features • Full Control over the development process • Open-Source
Usage	Django is suitable for high-end technology companies like Instagram, Udemy,	Flask is suitable for companies and projects that want experimentation with the module, and architecture of the framework like

Coursera, etc.	Netflix, Reddit, Airbnb, etc.

Django vs. Flask: Which one is better?

You are now well aware of the concepts and differences between Python-based Flask and Django. These frameworks have their individual features and characteristics that set them apart in their functionalities and usage.

Now, to choose one framework you might also need to learn about the pros and cons of both of the web frameworks. So, let us look at the primary pros and cons of Django and Flask.

Flask: Pros and Cons

Pros/Advantages

- Adaptable to the latest technologies
- Independent framework enables experimentation with architecture, libraries.
- Suitable for small case projects
- Requires small codebase size for simple functions
- Ensures scalability for simplistic applications
- Easy to build a quick prototype
- Routing URL functions through Werkzeug makes the process easier.
- Hassle-free application development and maintenance.
- Database integration is easy
- Extensible and easy core system.
- The power of the framework lies in its minimalistic features.
- Flexible and allows full control access.

Cons/Drawbacks

- MVP(Minimum Viable Product) development process is slow.
- Not suitable for big applications or projects.
- Complex maintenance for intricate implementations or system updates.
- There is no in-built admin site for maintaining models, insert, update or delete records.
- Does not support a proper database system and lacks Object-Relation Mapping.
- Absence of a strong community for support and growth.
- Security is uncertain, with no function for user authentication or login.

Django: Pros and Cons

Pros/Advantages

- The process of setting up and running the framework is easy and quick.
- Suitable and easy user interface for administrative functionalities.
- The built-in internationalization system enables the creation of multilingual websites.
- Integrated unit testing for the web application
- Support dynamic HTML pages
- In-demand framework amongst top tier companies.
- Easy and highly developed documentation
- Supports fully-featured Administration Interface
- Maximised scalability with less cost of hosting services
- Highly secured framework
- It is used for rate-limiting API requests from a single user.
- Assist you to define models for the URLs in your application

- Assist you to define models for the URLs in your application
- Ensures rapid development with a strong in-built template design.
- The prospects are positive and certain.

Cons/Drawbacks

- Monolithic working style making things too complicated and fixed.
- Prior knowledge of the framework is necessary.
- Codebase size is relatively larger.
- Too many functions and a high-end framework for a simple project.
- Profoundly based on Django ORM
- URL dispatching via controller reg-ex complicates the codebase.

Conclusion

Finally, we have arrived at the juncture of which one is better.

Django vs. Flask: One is an open-source framework for rapid development while the latter is a light-end framework for standard functionalities. Django and Flask are types of frameworks written in the Python programming language. These Python-based frameworks are considered to be one of the popular frameworks for web development, according to the Developers Survey 2018.

After reading and understanding the in-depth details about both of the web frameworks, one must easily conclude that both have their functionalities. It means that there must be a reason why both are among the popular Python-based frameworks in the domain of web development.

Flask renders full control and is highly suitable for small projects

that necessitate experimentation.

Django is complicated and requires vast knowledge but it stands out as one of the best frameworks for building sophisticated applications.

You could begin your learning in frameworks with Flask but upskill yourself in intricate tools and development with Django. Both of them are necessary skills for any web developer. Having fundamental knowledge and understanding of Python Flask and Django can put you on the map prior to other candidates while applying for a job.

Therefore, choose whatever you want but master it like a pro because they have a surging demand and are indispensable to the industry of web development.

FAQs

Q.1: Is Flask easier than Django?

Ans: Yes, the Flask learning curve is much simpler than Django.

Q.2: Which is better Django or Flask for beginners?

Ans: For beginners, Flask is recommended. It is easy to learn and used for making small applications that give room for experimentation and full control access to the development process.

Q.3: Is Django front end or back end?

Ans: Django is a full-stack web framework that allows suitability for both back-end and front-end development.

Q.4: Why is Flask preferred over Django?

Ans: The built-in libraries that come with Django do not give developers full control over the modules and functions that it provides. The Flask platform gives developers full control over application creation without requiring external libraries. Flask has a diversified working style. Currently, Django does not support virtual debugging.

Additional Resources

- Django Interview Questions
- Node.js vs Django
- Django Projects
- Django Books

Django Flask

PREVIOUS POST

SALARIES

Data Scientist Salary in India – For Freshers & Experienced

September 11, 2023

NEXT POST

COMPARE

Difference Between Alpha and Beta Testing

September 12, 2023

Categories

COMPARE 8 minute read

Difference Between Alpha and Beta Testing

September 12, 2023

Tools

Types

Table Of Contents [show]

Introduction

Consider a situation where you and your team developed software with lots of effort and hard work but after you delivered it to the client it contained loopholes and this happened due to a lack of testing that needs to be done for every software. It is important to perform various kinds of testing for your product to gain customer satisfaction and check software adaptability and functionality. Testing is the process of checking a system against various requirements that it is supposed to perform. It is an important step to be performed to ensure the quality of the product to be delivered. There is a variety of testing that is performed to achieve a quality product. Different types of testing are:

- User Acceptance testing
- Black Box testing
- End to End-testing
- Functional Testing
- Interactive Testing, and many more.

In this blog, we are going to discuss the two important types of testing, Alpha and Beta testing that come under User Acceptance testing. Both the testings are done on already tested products to get the real flavour of how the particular product will be used by real users. These two tests are based on the feedback of real users and different teams and it is achieved through different functionalities and techniques.

What is Alpha Testing?

It is one type of user acceptance testing. Its aim is to identify all

It is one type of user acceptance testing. Its aim is to identify all the possible issues and errors that a product can generate and resolve all those issues before the final product is delivered to the end-user or to the public. This testing got its name alpha because it is performed at an early stage, close to the end of the software development life cycle but before beta testing.

This kind of testing is performed by internal employees of the company that includes testers and Quality Analysts (QA) people. It is end-to-end testing done in a testing environment that makes sure the software developed meets all the expectations and works according to the requirements discussed at the start of development.

Alpha testing is carried out in two phases. The first phase is performed by software developers of the organization using some debugging software or tools. While the second phase is carried out by the testers or quality analysts. This phase covers black and white box testing also. Below is the process followed in alpha testing:

- Functional requirements and design specifications are reviewed.
- Developing test cases and test plans.
- Implementing those test plans.
- Defects raised from those test plans are then logged and informed to the developers.
- Developers resolve the defects raised and perform the testing once again.

Key Features of Alpha Testing

- Early identification of bugs in the design and functionality of the software.
- Imitation of real-time user environment.
- Helps in achieving better insights into software reliability at an early stage of development.
- More involvement of customers in the development process.
- Alpha Testing is performed at the developer’s site.

What is Beta Testing?

Beta Testing is also a type of acceptance testing that is done by a particular group of real users of the application. As it is the final stage of testing, it is done in production or a real environment. To achieve this type of testing the software is released to a few external members or customers who are not from the organization, they provide their feedback on the product's design, functionality, and overall quality.

This testing is highly beneficial as we are getting direct reviews from the person who is actually going to use the product after its final release and it decreases the failure risk of software. There are various types of Beta testing like traditional beta testing, public beta testing, technical beta testing, and focused and post-release beta testing.

Key Features of Beta Testing

- No requirement for a testing environment.
- Beta testing checks for reliability, robustness, and security of software.
- It is performed in the end-user's location.
- It is performed by real users outside the company.
- Beta testing commonly uses black box testing.

Key Differences: Alpha Testing vs. Beta Testing

- Alpha testing requires a testing environment whereas Beta testing doesn't require any such environment.
- Alpha testing uses both black and white box testing while Beta testing uses only blackbox testing.
- Alpha testing is done by testers and quality analysts inside the organization whereas Beta testing is done by real users who will be actually using the software

will be actually using the software.

- Alpha testing takes a longer duration to complete execution while Beta testing gets completed within a few weeks.
- Alpha testing is done at the developer's site while Beta testing is done at the client's site.
- Alpha testing does not check the security and reliability of the product while Beta testing checks for the security and reliability of the product in depth by the end users.
- Multiple test cycles are organized in alpha testing while in beta testing only one or two test cycles are there.

Difference Between Alpha and Beta Testing: Tabular Difference

Following are some of the differences between alpha and beta

Following are some of the differences between alpha and beta testing:

Alpha Testing	Beta Testing
This testing is performed by the employees of the organization	This testing is done by clients who are not part of the organization.
This kind of testing requires a specific environment for testing.	This does not require any environment for testing.
Robustness and security test is not performed in alpha testing	These parameters are checked during beta testing.
It is performed before the product launches into the market.	It is performed at the time of product marketing.
It performs many cycles to complete the testing. This may vary with the number of issues found.	It performs 1-2 cycles to complete the testing. This may vary with the user's feedback.
The main goal is to evaluate the quality of the product	The main goal is to evaluate customer satisfaction.
Both white-box and Black-box testing are involved.	It only involves black-box testing.
Activities can be controlled since it's performed on the developer's site.	Activities can't be controlled, since it's performed in the real environment.
This testing is done by highly-skilled employees. They have knowledge about	This testing is done by the end-users. They don't have the technical knowledge of the

the software product.	software product.
Stakeholders are the product management team, quality assurance team, and engineers.	Stakeholders are the product management team, user experience team, and quality management team.
Developers can resolve the bugs in alpha testing after testers inform them.	The feedback collected from the users is implemented in the future or in the next version of the application.

Entry and Exit Criteria of Alpha Testing

Entry Criteria:

Entry criteria define the conditions that must be met before the testing is started. The following are mentioned for alpha testing:

- All the features of the software product are implemented and testable.
- No urgent bugs are present in the software product.
- All features are tested on the primary platforms.
- QA build is ready for execution.
- The testing team has good knowledge of the software application.
- The test environment is properly set up.

Exit Criteria:

Exit criteria define the conditions that must be met to complete the testing. The following are the exit criteria for alpha testing:

testing. The following are the exit criteria for alpha testing:

- Creating the report of any kind of serious bugs.
- Notifying the bug-fixing issues to the developers.
- Serious issues are fixed and closed.
- Not to include any additional features.
- Delivery of test summary report.

Entry and Exit Criteria of Beta Testing

Entry Criteria:

- Positive results from alpha testing.
- Environment ready to release applications in public.
- The bug in alpha testing has been addressed.
- Beta sites are ready for installation.
- Regression testing (software testing done to confirm if the recent changes have not affected the existing features) of the bug fix has been completed.

Exit Criteria:

- All kinds of bugs (major and minor) have been fixed.
- A report of all serious bugs has been created.
- Feedback reports from the public are ready.
- Notifying the raised issue to developers.

Pros and Cons of Alpha Testing

The following are the advantages of Alpha testing:

- This testing reveals the bugs that were not noticed during

- This testing reveals the bugs that were not noticed during previous tests.
- It provides a better view of software reliability in the initial stages.
- It mimics the real-time user behaviour before going to beta testing.
- Serious errors are quite easily detected and resolved during this phase as testing is done at the developer's site.
- This testing ensures that users will get a better experience while using the application.
- It analyzes the risks before the product launch.

The following are the disadvantages of Alpha Testing:

- Some functionalities get missed for testing as the software is still in the development phase.
- The difference between the tester's perspective and the customer's perspective can cause discrepancies in software functioning.
- Only business requirements are covered in alpha testing.

Pros and Cons of Beta Testing

The following are the advantages of Beta Testing:

- Improves the quality of the software with the end-user's feedback.
- This testing increases customer satisfaction.
- Corner test cases are covered in this testing as applications are tested with all the possible cases by the users.
- It covers up the bugs that are revealed in previous tests.
- It is an opportunity to hand over the application to a few users before handing it over to the general public.

The following are the disadvantages of Berta Testing:

- Not all the participants assure you of the quality test.
- Reviewing the end user's feedback is high.
- Not all the reviews are beneficial for the software product.

Conclusion

Testing is a crucial step for any of the applications. It is important to perform both the testing i.e. Alpha and Beta with proper and well-defined procedures to make the application a successful product to use. In the IT world, the software is not worth it unless it is tested by the developers and liked by the end users.

FAQs

Q.1: What comes first, Alpha or Beta?

Ans: The testing that is performed first is Alpha testing. This testing is performed by the internal people of the organization. After this comes the Beta testing that is performed by the real-world users and on the basis of the usage, they provide their feedback.

Q.2: How long does alpha testing take?

Ans: Typically, alpha testing takes 1-2 weeks per test cycle based on how many issues are discovered and how many new features are released. It's not uncommon for the total Alpha phase to be 3-5x the length of the following Beta phase.

Q.3: How long does beta testing take?

Ans: Developers can spend **1-3 weeks** on beta testing depending on the size of the app. The typical duration of a Beta Test will vary depending on its objectives.

Q.4: Who is responsible for alpha testing?

Ans: Alpha testing is done to check whether the product meets the business requirements. This is done by the internal employees of the organization in a testing environment. They are highly skilled in the technical knowledge of the application.