Choosing the best Python web framework for beginner according to experienced users

Dejan Viduka*, Boris Ličina* and Luka Ilić*
* Faculty for Applied Management, Economics and Finance, University Business Academy, Belgrade, Serbia.
dejan@viduka.info, boris.licina@mef.edu.rs and luka.ilic@mef.edu.rs

Abstract - The world of programming and the range of programming languagesused is very large, and the dynamics of developing new solutions further complicate the selection of the right solution for their needs and knowledge development. The aim of this paper is to help beginners to choose the appropriate framework for learning and further development with the help of experienced users. With this choice, the users experience is very important - the framework that the user chooses can be used later in further development. This selected framework helps the user to dedicate himself exclusively to the solution and to get the most out of it and himself, rather than wandering between different solutions returning each time to the beginning of mastering a new tool. The paper presents all eight frameworks that have been presented and, based on research, offers a recommendation of experienced users that the solution chooses for beginners, while allowing an easier start and advanced web development.

I. INTRODUCTION

Nowadays, there are many tools used to develop web content. One of the tools that have found its application in the world of programming is the Python programming language [1]. Its application is large and is used from artificial intelligence to web development [2]. Many different libraries are used for the web, allowing users to develop from simple to very complex web solutions [3]. Such libraries are called frameworks that facilitate work for the specific purpose for which they were created [4]. Software frameworks are a set of standardized libraries [5] and programming tools that help create software applications.

The Python programming language itself has many such frameworks that allow users to work [6]. This way of developing various frameworks in a programming language is frequent, so it can be seen in the programming language PHP. In addition to so many possibilities, the user is faced with a big question, and that is the choice of the right framework for the needs of development, but also for the level of knowledge of the user who should use during development [7]. In this paper, we describe several frames that have been selected according to the user's choice. The aim of this paper is to show the attitudes of users and thus make it easier for users in practice to more easily and quickly choose one of the frameworks to use in their development.

Finding the frame quickly and easily facilitates the user and learning, and thus greatly affects the quality and speed of the final product. This is very important when it is known that a large number of frames are in circulation and that new ones appear every day that offer with them novelties that need to be mastered. Constant learning is something that is expected from a programmer, but it should also be reduced to an optimal measure. In order for a programmer to be good at his job, it is necessary to dedicate himself to a solution and master all the finesse that makes him a good programmer. If a programmer is constantly learning new technologies, then he has less time to dedicate himself more seriously to one of them and to get the most out of it. New programming languages and frameworks always bring some novelties that were once, but not always, better, so there are a large number of old programming languages that are still used today because they have quality solutions.

II. PYTHON WEB FRAMEWORKS

Python programming language is very modern in this respect and it develops a large number of frameworks that can be used for various purposes [8]. In this paper, we deal with web frameworks that compensate for the "flaws" of Python itself when it comes to web programming. In essence, we divide them into frames with which we get everything we need (full-stack) and those with which we can develop many necessary functionalities (micro-frame) [9] and thus do what we needed that frame for. This trend is very common in programming languages, with JavaScript and PHP leading the way, but Python is also present [10].

Interestingly, these two mentioned programming languages are mainly based on web development, while Python is generally usable in other areas and the framework allows it to be applied on the web [11]. The following are web frames created from the Python programming language.

A. Django

The Django framework was created in 2003 and is based on the Python programming language. Its application is in web development where it is successfully used by a large number of users connected in a community that provides customer support. Django supports both the older 2.x and newer 3.x versions of the Python programming language. To use this framework, you need a server environment as well as some of the supported databases, which are: PostgreSQL, MySQL, Oracle and SQLite. The user who decides to use Django in

his work has all the necessary tools for the development of a modern web project, regardless of its complexity in development. Its packages also include an administration panel that makes it easier for the user to work with the content. One of the great advantages of this framework is its support by the large community and well-done documentation. This documentation and support in the work enables an easier start as well as advanced development of modern projects. Of course, in addition to many advantages, Django also has its drawbacks, and one of them is that it is monolithic, which makes it poorly adaptable and flexible for newer user requirements. Despite this drawback, this framework is very suitable for the development of standard applications.

B. Pyramid

Pyramid is a web application framework developed as part of the Pilons project. It is an open source framework published under a BSD license. The development of this framework was inspired by libraries such as: Zope or Django. One of the advantages, and on the other hand for some it can be a disadvantage, is that this framework comes without predefined libraries. This means that this framework has more flexibility when it comes to adding new features by users. This framework, like Django, can work on both the old 2.x and the new 3.x version of the Python programming language. In addition to the many similarities, Django and Pyramid differ greatly in the possibility that users themselves develop new functionalities that the authors of the framework themselves did not anticipate. In addition to this, there is a big difference that there is no integrated support for databases. This just means that the user has to integrate it, but of course it is possible. This framework is very open which often increases development costs due to the need to independently develop the necessary tools. This makes this framework suitable for very experienced developers, but on the other hand it makes it very difficult for beginners to work with.

C. Bottle

Bottle is a fast, simple and lightweight micro web framework created using the Python programming language. It is distributed as a single file and has no dependencies other than the Python standard library, but not many features like other Python frames. One of its main advantages is the process of distributing a single file, which facilitates the sharing and distribution of applications. To start programming, all you need to do is download bottle.py and place it in the project folder. Its design is flexible, easy to use and facilitates the development of a multitude of simple web applications. However, due to its single-file distribution pattern, the framework is best suited for smaller applications, or prototypes, rather than larger projects.

D. CherryPy

This framework appeared in mid-2002 making it one of the oldest frameworks for the Python programming language. Despite this fact, it is less known among users than, say, Django and Flask. This frame is also a micro frame that is flexible and expandable by the user. It has built-in tools such as sessions, authorization, caching, routing, and database support. This framework allows developers to design and implement a project similar to or similar to other object-oriented Python projects. This fact allows for smaller source code, which results in rapid development but also optimal performance. One of its dominant advantages is the ease of configuration. This framework also supports the old 2.x and new 3.x versions of the Python programming language.

E. Tornado

Tornado is one of the Python web frames that focuses more on the network (asynchronous network library) and speed, including all the features of the web frame. Not only does it come with performance-enhancing features, but it also has features like stenciling, routing to build a simple to scalable web application. By using non-blocking network I / O, the Tornado can scale to thousands of open connections, making it ideal for long-term testing and applications that require a long-term connection with each user. The tornado runs on all Unix platforms as well as Linux and BSD.

F. web2pi

This web framework was created in 2007, and its real purpose was to serve as a teaching tool in working with students. This means that it was developed from the beginning to be easy to use. Its creators were inspired by the Django frame but also by Ruby on Rails. This web framework contains a web server that enables SSL and streaming, work with relational databases as well as an integrated development environment based on the Internet as well as a batch management interface. Support in working with this framework is reflected in the extensive documentation that the user is gradually introducing into the issue. In addition to all its advantages, the community of this web framework is significantly smaller than Django or Pyramid-e, which in practice means that less support can be expected from experienced users.

G. Zope

This Python framework originated as an open source social project that is object oriented. Its full name is “Z Object Publishing Environment” or ZOPE for short. This framework is known for making Python known around the world. Zope has a developed community that in the last few years has developed additional frameworks designed for different uses and using different development principles, while retaining the old philosophy. Some of these frameworks that have been developed are: Plone, BlueBream and Grok. Zope is a leading Open Source Application Server and Content Management Framework, specializing in content management solutions, portal content management and custom applications. Zope is managed by the global Zope community, with thousands of developers and companies around the world. Zope enables teams to collaborate in creating and managing dynamic Internet-based business applications, such as intranets and portals. The Zope web application server and its Content Management Framework form the basis for one of the most popular and powerful content management systems in the world - Plone.

H. Flask

This framework is a micro web application created in the Python programming language. It is published under the BSD license. Some of the well-known applications created using this framework are LinkedIn and Pinterest. This micro web application comes with much basic functionalities that can be easily extended. The motto of this framework is "one drop at a time", and this is achieved through extensive documentation that gradually introduces the user to the issue. This is one of the virtues that beginners opt for this framework at the very beginning, but they also often remain faithful to it because of its capabilities in working with very complex applications. In addition to its advantages, this framework also has disadvantages that are reflected in a large number of add-ons that are often not updated. In such cases, users have to spend a lot of time looking for replacements that have similar functions and are still active and supported. It is almost always difficult to find documentation or instructions for use for replacement files, so in such cases the help of a large community is necessary.

III. CHOOSING THE BEST PYTHON WEB FRAMEWORK

The choice of the appropriate framework depends on the domain for which it is planned to be used. Practically, this means that there is no perfect solution, but it is a matter of user choice that depends on the purpose for which it will be used. The choice of users can depend on several different factors such as, ease of use, support and even ease of setting up the user environment.

Some of these frames are also divided into two main categories, full-stack frames and micro-frames (sometimes called non-full-stack). By choosing a full-stack framework, you will have at hand a variety of tools and packages for developing any type of application. These are MVC architecture, ORM, routers, security and more.

In this work, we learned from users who use Python what they prefer in various web development frameworks. We included in the research eight frameworks that our respondents used in their work. We asked users three piranhas regarding ease of use, quality of support, and ease of setting up the user environment from the frameworks they used or prefer to use. For these three questions, users answered us using ratings that ranged from 1 to 5, which we tried to make it easier for them to rate. After these questions, we calculated the average grade and presented it on a scale from 1 to 5.

The obtained results are shown in Table 1, in which the individual ratings can be seen as well as the final ocean for each frame according to the user's opinion. The table also shows the number of users who rated each box. Each user could rate only one framework he used and rate it. 222 developers participated in this research, which used some of the given frameworks in their work.

From the presented results, the table shows the percentage of users who voted for one of the frames. It is clear that the largest number of users used Django, followed by Flask. It is also interesting that Flask is the first in the received grades, followed by Django and the others.

TABLE I. PRESENTATION OF RESEARCH RESULTS AMONG EXPERIENCED DEVELOPERS

id	Frameworks	Usability	Support	Adjustment	Participants	Mark
1.	Django	4.2	4.1	4.0	121	4.1
2.	Tornado	3.9	3.7	3.6	20	3.7
3.	Bottle	4.1	3.8	4.0	12	4.0
4.	web2py	4.2	4.1	3.4	12	3.9
5.	CherriPy	3.9	3.2	3.5	12	3.5
6.	Pyramid	3.6	3.7	3.4	13	3.6
7.	Zope	3.5	3.4	3.9	4	3.6
8.	Flask	4.6	4.2	4.3	28	4.4

IV. DISCUSSION

There are a large number of frames on the market that serve a similar purpose, but each is different. These differences are especially visible in beginners who are usually very confused and find it difficult to decide on a solution or experiment with everyone. Experienced users have already made a choice of frames that they use more often, which can be seen from our research. All frameworks listed here are used, but each has its own purpose, as well as the user to whom it suits, depending on the knowledge, experience and project for which they use it. Our research confirms what is known among experienced users and can be read on expert forums dedicated to web development and Python. The table clearly shows the number of participants who in large numbers opted for the two leading frameworks, namely Django and Flask. Other frames from our table also have good grades, but they are still used by a smaller number of users, so they recommend them to others, especially beginners.

According to the ratings, two frames stood out, namely: Flaski, Django and Bottle. These ratings were obtained based on the three parameters we used, namely: support, usability, and tuning. These parameters are very important for every user, from professionals to beginners who first encounter these frameworks for the realization of their projects.

What is characteristic of all the frameworks from our research is that they were created using the Python programming language and therefore require users to know that programming language. Another similarity is that they are all intended for web development and generally differ in their purpose and the size of the community that supports them. Community size is very important in open source projects [12] as well as documentation that helps to easily master each individual programming language or in this case the framework. Three separate frames from our research cover almost all segments that we have listed, so it is no wonder that they are used so much and of course recommended to users from the very beginning to more advanced development. This problem helps users to decide from the start on a

framework that can later serve them for advanced development, depending on their knowledge and needs.

V. CONCLUSION

There are many frameworks on the market that meet most of the needs of our users, and we usually leave the choice to the users themselves. One way of choosing is experiential, so in this research we asked professionals to recommend a framework according to their opinion that beginners should consider for their work. The research did not bring a ready-made solution which framework is better than someone else's, but showed what other users recommend for work. The paper singled out two frameworks according to the grades they received, but also according to the number of users who recommended them. Django is definitely the first choice of the user, which is quite logical considering its application, documentation and the support that the community provides to it. The second frame is Flask and it is the first in terms of ratings, but a smaller number of users recommend it compared to Django. Flask is also very useful, which can be seen from the listed projects for which it is used in the world, but users still recommend Django first and only then Flask.

The choice is up to the user, and this work and research in it has managed to reduce the eight offered frameworks to two from which users can choose, again depending on the need, knowledge, support or some subjective feeling. Each of these frameworks is a good solution and as such should be viewed, but the user's experience has given its recommendation and as such should always be taken into account when choosing and deciding on the framework that the user chooses for learning and further development.

REFERENCES

[1] Ruba Abu Khurma, Ibrahim Aljarah and Ahmad Sharieh () EvoloPy–FS: An Open–Source Nature–Inspired Optimization Framework in Python for Feature Selection, School for Information Technology, The University of Jordan,Amman,Jordan.

[2] Hina Afzal, Arisha Kamran, Asifa Noreen (2020) Survival Analysis of Python and R within the Job Market Trend, Journal of Information Technology and Computing, 2020 Volume 1, Issue 1 : 31 – 40, DOI : 10.48185/jitc.v1i1.94

[3] Rašić Nino (2018) Web alikacije za prikupljanje i analizu meteroloških podataka, University of Pula, Croatia, Master's thesis.

[4] Vangala Rama Vyshnavi, Amit Malik (2019) Efficient Way of Web Development Using Python and Flask, Vangala Rama Vyshnavi et al. International Journal of Recent Research Aspects ISSN: 2349-7688, Vol. 6, Issue 2, pp. 16-19.

[5] Devndra Ghimire (2020) Comparative study on Python web frameworks: Flask and Django, Metropolia University of Applied Sciences, Bachelor's Thesis.

[6] Vogel, P., Klooster, T., Andrikopoulos, V., & Lungu, M-F. (2017). A Low-Effort Analytics Platform for Visualizing Evolving Flask-Based Python Web Services. 109-113. Paper presented at 2017 IEEE Working Conference on Software Visualization, Shanghai, China. https://doi.org/10.1109/VISSOFT.2017.13

[7] Barbaresi, A., (2020). htmldate: A Python package to extract publication dates from web pages. Journal of Open Source Software, 5(51), 2439. https://doi.org/10.21105/joss.02439

[8] K. R. Srinath (2017) Python – The Fastest Growing Programming Language, International Research Journal of Engineering and Technology (IRJET), Volume: 04 Issue: 12.

[9] Kameryn Larkins (2020) A Comparison of Python Web Development Microframeworks, Southern Connecticut State University, New Haven, Connecticut, Master of Science.

[10] Marija Ševković (2020) Mreža mikroservisa sa distribuiranom bazom podataka u okruženju Flask, Matematički fakultet, Univerzitet u Beogradu, Master's thesis.

[11] Karnalim, O., & Ayub, M. (2017). The Use of PythonTutor on Programming Laboratory Session: Student Perspectives. Kinetik : Game Technology, Information System, Computer Network, Computing, Electronics, and Control, 2(4). doi:http://dx.doi.org/10.22219/kinetik.v2i4.442

[12] Hongyu Zhai, Casey Casalnuovo and Prem Devanbu (2019) Test Coverage in Python Programs, MSR '19: Proceedings of the 16th International Conference on Mining Software RepositoriesMay 2019 Pages 116–120https://doi.org/10.1109/MSR.2019.00027