

# Comparative analysis of web applications implemented in: PHP and Python

## Analiza porównawcza aplikacji webowych napisanych w językach: PHP oraz Python

Jakub Zborowski\*, Maciej Pańczyk

*Department of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland*

### Abstract

This article presents a comparative analysis of two web applications implemented in PHP and Python. Test applications were created and equipped with the same functionality used in tests consisting in measuring the server response times to INSERT, SELECT, UPDATE and DELETE requests - handling database operations. The purpose of the research was to compare both languages in terms of selected criteria. Their performance, source code volume and popularity were compared.

*Keywords:* web application; performance analysis; PHP; Python

### Streszczenie

Artykuł przedstawia analizę porównawczą dwóch aplikacji webowych napisanych w językach: PHP oraz Python. Stworzono aplikacje testowe, które zostały wyposażone w tę samą funkcjonalność wykorzystaną w badaniach polegających na pomiarze czasów odpowiedzi serwera na żądania typu INSERT, SELECT, UPDATE i DELETE – obsługujące operacje na bazie danych. Celem badań było porównanie obydwu języków pod względem wybranych kryteriów. Porównywano ich wydajność, objętość kodu źródłowego oraz popularność.

*Słowa kluczowe:* aplikacja webowa; analiza wydajności; PHP; Python

\*Corresponding author

Email address: [j\\_zborowski@o2.pl](mailto:j_zborowski@o2.pl) (J. M. Zborowski)

©Published under Creative Common License (CC BY-SA v4.0)

## 1. Wstęp

Aplikacje internetowe w dzisiejszych czasach stają się coraz bardziej popularne wśród użytkowników co powoduje, że powoli wypierają one z naszego życia aplikacje desktopowe.

Bardzo prędko rozwijająca się technologia umożliwia wykorzystanie dużej ich elastyczności. Aby móc korzystać z aplikacji internetowej, wymagane jest wyłącznie posiadanie przeglądarki – dzięki temu ignorujemy konieczność pobierania i instalowania programu na każdym z urządzeń, przez co zyskujemy wygodny dostęp do jego funkcji, bądź plików zarówno z telefonu, tableta jak i komputera. Pozwala to na szybki wgląd do plików oraz przesyłanie wiadomości z dowolnego miejsca. Aplikacje desktopowe borykają się również z koniecznością systematycznych aktualizacji - nie doświadczymy tego rodzaju problematycznych przerw w przypadku korzystania z wersji webowych. Warto pamiętać też o wymaganiach systemowych, koniecznych do zainstalowania oprogramowania desktopowego. Biorąc pod uwagę aplikacje internetowe wszystko, czego potrzebujemy, to przeglądarka, możemy więc korzystać z jej funkcji bez konieczności zamartwiania się, czy nasz sprzęt ma odpowiednie parametry, aby poradzić sobie z wymaganiami programu. Wszystkie te zalety składają się na dużą popularność aplikacji internetowych.

Dodatkowo pisząc aplikację możemy użyć szkieletów programistycznych, co znacząco ułatwi proces

tworzenia. Zaprojektowano je w taki sposób, aby zapewnić elastyczność, mają one również dobrą logikę wewnętrzną, którą narzucają stworzonej aplikacji. Korzystając z tego rozwiązania programista pisze mniej kodu, a szkielety programistyczne same w sobie są dobrze zaprojektowane i przetestowane – co czyni je niezawodnymi.

## 2. Przegląd literatury

Języki PHP oraz Python są językami, które zajmują czołowe pozycje w zakresie tworzenia stron i aplikacji internetowych. Dla języka PHP jednym z najbardziej popularnych i profesjonalnych szkieletów programistycznych jest szkielet Symfony, natomiast dla języka Python czołowym, a jednocześnie dosyć starym szkieletem jest szkielet Django. W przypadku aplikacji internetowych bardzo ważną kwestią jest ich jakość i związana z nią wydajność. W związku z tym testowanie aplikacji tego typu jest skomplikowanym oraz pracochłonnym procesem.

W artykule [1] autorzy Kai Lei, Yining Ma oraz Zhi Tan porównali i ocenili technologie służące do budowy aplikacji internetowych: PHP, Python i Node.js. Badania zrobiono, wykonując identyczne testy na aplikacjach posiadających te same funkcjonalności, ale zbudowanych przy użyciu tych trzech różnych technologii. Testy aplikacji wyświetlającej tekst „Hello World!”, obliczających i zwracających dziesiąty, dwudziesty i trzydziesty wyraz ciągu Fibonacciego, jak i przeprowadzających

operacje na bazie danych, w większości przypadków wykazały, że w stosunku do pozostałych testowanych technologii największą wydajnością cieszy się środowisko Node.js.

W artykule [2], autorzy Giuseppe Antonio Di Lucca oraz Anna Rita Fasolino podkreślają istotność testowania aplikacji pod względem wydajności przy zastosowaniu różnych obciążeń osiąganych poprzez zwiększenie liczby zapytań oraz wysyłanie różnej wielkości danych. W tego typu badaniach podstawową wykorzystywaną miarą jest czas wykonania poszczególnych czynności i odpowiedzi serwera.

W artykule [3], autor Klaus Purer porównał technologie służące do budowy aplikacji internetowych: PHP, Python oraz Ruby. Na początku została porównana semantyka i składnia, następnie pomocne cechy, takie jak: wyłapywanie błędów, abstrakcja relacyjnej bazy danych czy funkcjonalne cechy języków. Autor również porównuje języki pod względem bezpieczeństwa i finalnie również wydajności. Po przeprowadzonych badaniach okazało się, że jedynie w obszarze popularności i dostępności PHP był na pierwszym miejscu, a w kwestii wydajności wszystkie trzy technologie były na równi. W pozostałych, tj.: czytelność, użyteczność, bezpieczeństwo, abstrakcja bazy danych, wyłapywanie błędów oraz funkcjonalne cechy - Python delikatnie zwyciężył nad językiem Ruby, a język PHP zdecydowanie odstawał. Ostatecznie autor podkreśla, że wybór języka jest kwestią indywidualną, a dyskusje o językach są emocjonalne i irracjonalne.

Dotychczas żaden z badaczy nie pokusił się o porównanie popularności języków PHP oraz Python na podstawie liczby wyszukiwań, co zostało przeze mnie zrobione. Dodatkowo w wyżej wymienionych artykułach wykorzystane zostały starsze wersje języków.

### 3. Porównywane języki

#### 3.1. PHP

Język PHP (ang. Hypertext Preprocessor) to skryptowy język programowania działający po stronie serwera, który został stworzony w roku 1994 przez Rasmusa Lerdorfa [4]. W rankingu popularności języków programowania TIOBE [5] z sierpnia 2022 roku, zajmuje on całkiem wysokie dziesiąte miejsce.

W zakresie zastosowania na stronach internetowych język PHP jest w ścisłej czołówce. Wedle ankiety przedstawionej w serwisie W3Techs język PHP jest używany przez 77.4% stron, dla których język programowania po stronie serwera jest znany [6]. Dodatkowo PHP jest wykorzystywany na wielu popularnych stronach, np.: Facebook, Instagram, WordPress czy Wikipedia.

#### 3.2. Python

Język Python to język programowania wysokiego poziomu ogólnego przeznaczenia, którego pierwsza wersja ukazała się w 1991 roku, a jego twórcą jest Guido van Rossum [7]. Charakteryzują go takie cechy jak:

- jest w pełni obiektowy,

- umożliwia programowanie w różnych stylach (strukturalne, obiektowe, funkcyjne, itd.),
- jest językiem interpretowanym,
- wcięcia w kodzie są wykorzystywane do tworzenia bloków.

Obecnie jest on używany przez setki tysięcy programistów, co podkreśla jego miejsce w rankingu popularności języków programowania TIOBE z sierpnia 2022 roku, gdzie Python zajmuje pierwsze miejsce – tuż nad językiem C oraz językiem Java.

### 4. Metoda badań

#### 4.1. Opis eksperymentu

Analizę porównawczą aplikacji webowych, które zostały napisane w językach PHP oraz Python przeprowadzono na podstawie trzech scenariuszy badawczych, w ramach których wykonano:

1. pomiar czasów wykonywania się aplikacji służących do operowania na bazie danych, zmierzony za pomocą dostępnych funkcji,
2. pomiar objętości kodu aplikacji zaimplementowanych na rzecz danego polecenia, polegający na zliczeniu liczby linii kodu,
3. badanie popularności porównywanych języków poprzez sprawdzenie liczby ich wyszukiwań na wybranych platformach.

Badanie czasów wykonywania się aplikacji zostało podzielone na 4 główne operacje na bazie danych: INSERT, SELECT, UPDATE oraz DELETE. Każde z nich było badane w 5 scenariuszach różniących się liczbą rekordów, na których były wykonywane: 1, 10, 100, 1000 i 10000 rekordów. Dla każdego scenariusza aplikacja wykonywała się 10-krotnie, aby wyniki były bardziej wiarygodne. Następnie została wyliczona z nich średnia, która jest widoczna na rysunkach w kolejnym rozdziale.

Do generowania niezbędnych danych potrzebnych do operowania na bazie danych został wykorzystany pakiet Faker [8], który dla języka Python został stworzony na podstawie inspiracji biblioteką Faker, stworzoną dla języka PHP [9]. Przy wykorzystaniu odpowiednich funkcji umożliwiają one generowanie imion, nazwisk, liczb czy nawet adresów.

#### 4.2. Środowisko testowe

W tabeli 1 pokazano środowisko testowe, na którym przeprowadzono badania. Oprócz parametrów komputera znajdują się tu również wersje analizowanych języków programowania.

Według artykułu opublikowanego na stronie Hackr.io [10] przez Amana Goela pracującego w Indyjskim Instytucie Technologii w Bombaju, który zgromadził ponad 50 tysięcy obserwatorów na swoim profilu na portalu społecznościowym dla programistów - LinkedIn [11] - wersje 5.x języka PHP były wolne, jednakże nowsze wydanie, 8.1, czyni ten język zdecydowanie szybszym. Prawie 3-krotnie szybszym od typowego programu napisanego w języku Python.

Tabela 1: Specyfikacja urządzenia testowego

| Sprzęt            |                       |
|-------------------|-----------------------|
| Procesor          | AMD Ryzen 5 3600X     |
| Pamięć RAM        | 16,0 GB               |
| System operacyjny | Windows 10 Home 64bit |
| Dysk              | TOSHIBA HDWD120       |
| Oprogramowanie    |                       |
| PHP               | 8.1.6                 |
| Python            | 3.7.0                 |

### 4.3. Pomiar objętości kodu

W scenariuszu 2 zliczono liczbę linii kodu dla czterech porównywanych operacji na bazie danych, które w każdym z języków realizowały te same zadania. Wiersze zawierające nawiasy klamrowe lub puste wiersze zostały pominięte w pomiarach.

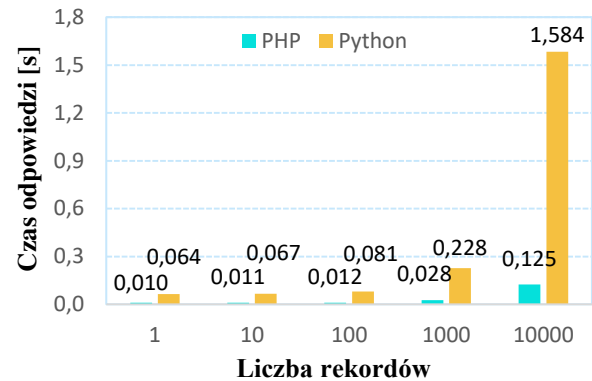
### 4.4. Badanie popularności

W scenariuszu 3 przeprowadzono badanie popularności języków poprzez sprawdzenie liczby ich wyszukiwań na trzech wybranych platformach. Pierwszą z nich jest najpopularniejsza wyszukiwarka – Google [12], która zazwyczaj jest wybierana jako pierwsza, gdy ludzie czegoś nie wiedzą – tutaj wzięto pod uwagę jedynie wyniki, które dokładnie posiadały szukane słowo. Jako drugą platformę wybrano jedno z najbardziej popularnych forów programistycznych - Stack Overflow [13], na którym brano pod uwagę jedynie wątki, które posiadały jakąkolwiek odpowiedź. Ostatnią platformą, która została wybrana jest GitHub [14] - hostingowy serwis internetowy przeznaczony do projektów programistycznych, będący obecnie najpopularniejszym hostem projektów open source. Dla GitHuba wzięto pod uwagę liczbę repozytoriów, aczkolwiek pod każdym innym względem również górował ten sam język.

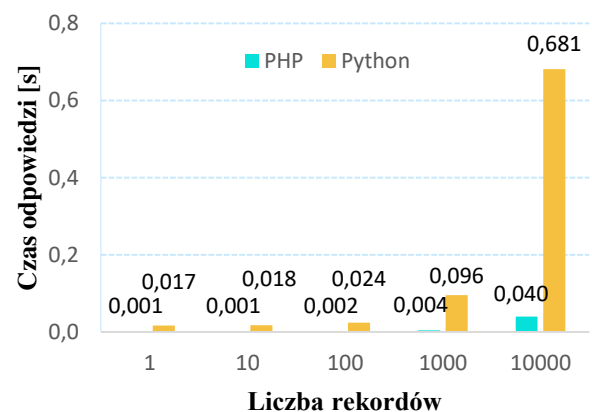
## 5. Wyniki

### 5.1. Pomiar czasów

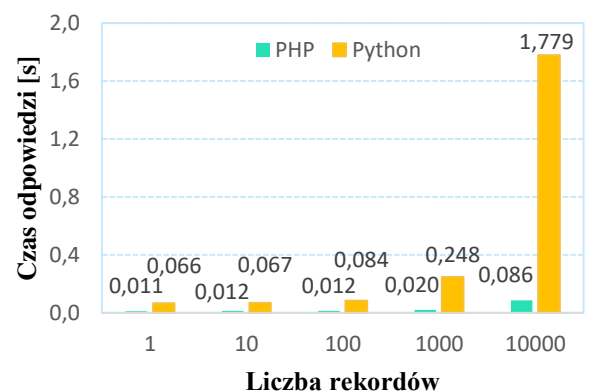
Na poniższych rysunkach przedstawiono zebrane wyniki dla scenariusza 1, w ramach którego przeprowadzono testy dla poszczególnych operacji na bazie danych: zapisu, odczytu, aktualizacji i usuwania. Zostały one wykonane na różnej liczbie rekordów, począwszy od 1, następnie 10, 100, 1000 oraz 10000 rekordów.



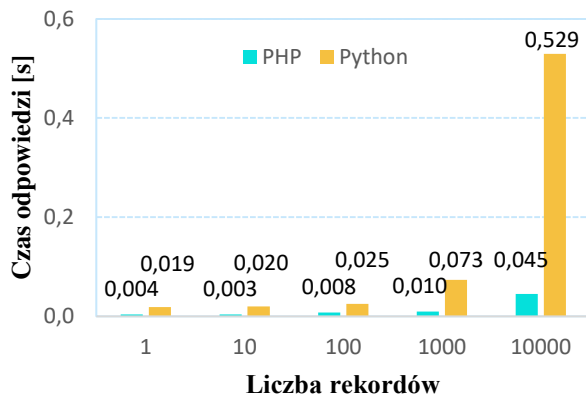
Rysunek 1: Czas wykonania zapytania INSERT [s].



Rysunek 2: Czas wykonania zapytania SELECT [s].



Rysunek 3: Czas wykonania zapytania UPDATE [s].



Rysunek 4: Czas wykonania zapytania DELETE [s].

Powyższe rysunki ukazują średnie czasy wykonywania się programów przy różnej liczbie rekordów, co jest równoznaczne ze zróżnicowanym obciążeniem.

Z wykresów tych widać, że przeskok pomiędzy 1, 10 oraz 100 rekordami dla każdego z języków jest stosunkowo niewielki, jednakże porównując je ze sobą widać zdecydowaną różnicę. Dla polecenia DELETE przy działaniu na jednym rekordzie PHP okazał się około 5 razy szybszy, biorąc pod uwagę polecenia INSERT oraz UPDATE wciąż tylko na jednym rekordzie jest to już 6-krotna przewaga na korzyść języka PHP. Gdy przejdziemy natomiast do polecenia SELECT przewaga rośnie dramatycznie i Python przy działaniu na jednym rekordzie jest już 17 razy wolniejszy.

Patrząc na różnicę pomiędzy 100 oraz 1000 rekordów dla języka PHP wynosi ona maksymalnie 16 milisekund - dla polecenia INSERT. Przechodząc do języka Python są to już różnice rzędu 147ms, 72ms, 164ms oraz 48ms dla poszczególnych poleceń. Porównując obydwa języki przy liczbie 1000 rekordów: przewaga na korzyść PHP jest 8-krotna dla polecenia INSERT, 24-krotna dla polecenia SELECT, 12-krotna dla polecenia UPDATE i 7-krotna dla polecenia DELETE.

Gdy spojrzymy na przeskok z 1000 na 10000 rekordów dla języka PHP: czas wykonywania się programu dla poleceń INSERT, UPDATE oraz DELETE zwiększył się około 4,5 raza, a dla polecenia SELECT 10-krotnie. Biorąc pod uwagę ten sam przeskok dla języka Python: dla każdego z poleceń wynosi on około 7-krotny wzrost.

Porównując wyniki przy 10000 rekordów: przy poleceniu INSERT język PHP okazał się ponad 12-krotnie szybszy, przy poleceniu SELECT 17-krotnie, przy poleceniu UPDATE ponad 20-krotnie, natomiast przy poleceniu DELETE jest to przewaga ponad 11-krotna.

## 5.2. Pomiar objętości kodu

Średnia liczba linii kodu przypadająca na implementację funkcjonalności w języku PHP wynosi: 19,75 (tabela 3). W przypadku języka Python średnia liczba linii kodu przypadająca na implementację funkcjonalności wynosi: 26,75 (tabela 4).

Tabela 3: Objętość kodu dla języka PHP

| Funkcjonalność | Liczba linii kodu |
|----------------|-------------------|
| INSERT         | 21                |
| SELECT         | 19                |
| UPDATE         | 23                |
| DELETE         | 16                |

Tabela 4: Objętość kodu dla języka Python

| Funkcjonalność | Liczba linii kodu |
|----------------|-------------------|
| INSERT         | 27                |
| SELECT         | 28                |
| UPDATE         | 27                |
| DELETE         | 25                |

Można zauważyć, że do zaimplementowania tych samych funkcjonalności język PHP potrzebował mniejszej objętości kodu. Dla języka Python było potrzebne o 35% więcej linijek kodu. Wynika to m.in. z tego, że w składni Pythona obowiązuje technika „wcięć”, a przez to niektóre instrukcje, możliwe do zapisania w innych językach programowania w jednej linijce trzeba rozdzielać na kilka.

## 5.3. Badanie popularności

W tabeli 2 przedstawiono porównanie popularności języków PHP oraz Python poprzez sprawdzenie liczby wyszukiwań tychże języków w wyszukiwarce Google, wątków z udzieloną jakąkolwiek odpowiedzią na serwisie społecznościowym Stack Overflow oraz liczby repozytoriów na serwisie internetowym GitHub.

Tabela 2: Porównanie popularności języków

| Język  | Rok wydania | Google      | Stack Overflow | GitHub    |
|--------|-------------|-------------|----------------|-----------|
| PHP    | 1994        | 844 000 000 | 1 275 000      | 851 000   |
| Python | 1991        | 712 000 000 | 1 722 000      | 2 708 000 |

Wyniki te pokazują, że PHP zgromadził więcej wyszukiwań w Google, jednakże Python posiada o wiele większe wsparcie społeczności, co nie powinno dziwić, gdyż jest on popularniejszym oraz bardziej uniwersalnym językiem.

## 6. Wnioski

Aplikacje internetowe stają się coraz bardziej popularne w obecnych czasach, zapotrzebowanie na nie wciąż rośnie, a im samym stawiane są coraz wyższe wymagania. Pojawiają się coraz to nowsze technologie do tworzenia tych aplikacji. Z uwagi na to programiści są zmuszeni do trudnego wyboru – jaką technologię wybrać do zaimplementowania swojej aplikacji.

Celem niniejszej pracy było porównanie dwóch popularnych języków programowania używanych do tworzenia aplikacji internetowych: języka PHP oraz języka

Python. Do przeprowadzenia badań przygotowano aplikacje testowe, które zostały zbudowane w podobny sposób, zgodnie z założeniami. Posiadały one identyczny zestaw funkcjonalności, te same dane, oraz były uruchamiane w jednakowy sposób. W analizie porównawczej wzięto pod uwagę trzy kryteria: popularność, wydajność oraz objętość kodu.

Biorąc pod uwagę uzyskane wyniki wydajności, można stwierdzić, że język PHP wypadł o wiele lepiej od języka Python. Przy pracy nad najnowszą, aktualnie dostępną wersją PHP programiści postarali się o zwiększenie szybkości jego działania i jak można zauważyć przyniosło to oczekiwany efekt.

Pod względem objętości kodu Python, który charakteryzuje się wyjątkowym sposobem pisania kodu, potrzebował średnio o 7 linijek więcej, co przez wspomnianą wcześniej technikę „wciąć” nie powinno dziwić. Implementacja aplikacji w języku Python była o wiele łatwiejsza, jednakże składała się z większej liczby linijek kodu źródłowego.

Porównując popularność badanych języków w wyszukiwarce Google zwyciężył PHP, jednak patrząc na wsparcie społeczności oraz liczbę projektów zdecydowanie przeważa ten drugi. Python jest o wiele bardziej uniwersalnym językiem stosowanym w wielu różnych dziedzinach technologicznych, podczas gdy PHP jest używany głównie do budowania skryptów po stronie serwera internetowego.

Wyniki przeprowadzonych badań wypadają na korzyść języka PHP. Wykonana analiza jest ograniczona, gdyż dotyczyła bardzo prostej aplikacji testowej, wykonującej jedynie cztery rodzaje operacji na bazie danych – zapis, odczyt, aktualizację oraz usuwanie. Aby przeprowadzić dokładniejszą analizę należałoby poszerzyć działanie aplikacji testowych i dodać kolejne kryteria do zbadania.

## Literatura

- [1] K. Lei, Y. Ma, Z. Tan, Performance comparison and evaluation of web development technologies in php, python, and node.js, In 2014 IEEE 17th international conference on computational science and engineering, IEEE (2014) 661-668.
- [2] G. A. Di Lucca, A. R. Fasolino, Testing web-based applications: The state of the art and future trends, Information and Software Technology 48 (2006) 1172-1186.
- [3] K. Purer, PHP vs. Python vs. Ruby–The web scripting language shootout, Vienna University of Technology, 2009.
- [4] L. Welling, L. Thomson, PHP and MySQL Web development, Sams Publishing, 2003.
- [5] TIOBE Programming Community Index, <https://www.tiobe.com/tiobe-index/>, [14.08.2022].
- [6] W3Techs – Technology Surveys, <https://w3techs.com/technologies/comparison/pl-aspnet.pl-php.pl-python>, [24.08.2022].
- [7] M. Lutz, Programming python, "O'Reilly Media, Inc.", 2001.
- [8] Faker – generator danych dla języka Python, <https://faker.readthedocs.io/>, [17.08.2022].
- [9] Faker – generator danych dla języka PHP, <https://github.com/fzaninotto/Faker/blob/master/readme.md>, [17.08.2022].
- [10] Hackr.io – artykuł Amana Goela, <https://hackr.io/blog/python-vs-php>, [18.08.2022].
- [11] LinkedIn – profil Amana Goela, <https://www.linkedin.com/in/goel-aman/>, [18.08.2022].
- [12] Wyszukiwarka Google, <https://www.google.com/>, [16.08.2022].
- [13] Stack Overflow – forum programistyczne, <https://stackoverflow.com/>, [16.08.2022].
- [14] GitHub – serwis internetowy, <https://github.com/>, [16.08.2022].