

Možnosti frameworku PyScript

Bc. Jan Sadílek

Diplomová práce
2023



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2022/2023

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Jan Sadílek**
Osobní číslo: **A21177**
Studijní program: **N0613A140022 Informační technologie**
Specializace: **Softwarové inženýrství**
Forma studia: **Kombinovaná**
Téma práce: **Možnosti frameworku PyScript**
Téma práce anglicky: **PyScript Framework Capabilities**

Zásady pro vypracování

1. Nastudujte a popište problematiku webového vývoje nejen v jazyce Python.
2. Rozepište aktuální možnosti frameworku PyScript a porovnejte je s konkurenčními možnostmi.
3. Navrhněte praktické ukázky s využitím frameworku PyScript.
4. Implementujte ukázkovou webovou aplikaci za použití frameworku PyScript.
5. Vytvořte praktický tutoriál pro framework PyScript.
6. Vše vhodně otestujte a popište.

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. ULLI, Stanley. Intro to PyScript: Run Python in the browser.LogRocket[online]. 2022. Dostupné: <https://blog.logrocket.com/pyscript-run-python-browser/>
2. JONES, Alan. Create an Interactive Web App with PyScript and Pandas.Towards Data Science[online]. 2022. Dostupné: <https://towardsdatascience.com/create-an-interactive-web-app-with-pyscript-and-pandas-3918ad2dada1>
3. LAWSON, Loraine. Pyscript: A Browser-Based Python Framework for the 99%.The New Stack[online]. 2022. Dostupné z: <https://thenewstack.io/pyscript-a-browser-based-python-framework/>
4. ZACZYŃSKI, Bartosz. A First Look at PyScript: Python in the Web Browser.Real Python[online]. 2022. Dostupné z: <https://realpython.com/pyscript-python-in-browser/>
5. E, Senthil. PyScript-Use Python Code in HTML.Medium[online]. 2022. Dostupné z: <https://medium.com/analytics-vidhya/pyscript-use-python-code-in-html-f7c8b49486a4><https://www.freecodecamp.org/news/pyscript-python-front-end-framework/>
6. What is Pyodide?.Pyodide[online]. Pyodide. Dostupné: <https://pyodide.org/en/stable/>
7. Say Hello to PyScript.PyScript[online]. Anaconda. Dostupné z: <https://pyscript.net/><https://webassembly.org/>
8. WebAssembly. WebAssembly[online]. Dostupné z: <https://webassembly.org/>
9. PECINOVSKÝ, Rudolf.Python: Kompletní příručka jazyka pro verzi 3.10. Grada, 2021, 1 online zdroj (608 stran). ISBN 978-80-271-4423-5. Dostupné také z: <https://www.bookport.cz/AccountSaml/SignIn/?idp=https://shibboleth.utb.cz/idp/shibboleth&returnUrl=/kniha/python-10436/>
10. CROWDER, T. J.JavaScript: the new toys. Hoboken, NJ: Wrox, a Wiley brand, 2020, 1 online resource. Dostupné z: doi:9781119367901
11. BROWN, Tiffany B., Kerry BUTTERS a Sandeep PANDA.HTML5 okamžitě: [ovládněte HTML5 za vikend]. Brno: Computer Press, 2014, 256 s.ISBN 9788025142967.

Vedoucí diplomové práce: **Ing. Petr Žáček, Ph.D.**
Ústav informatiky a umělé inteligence

Datum zadání diplomové práce: **2. prosince 2022**
Termín odevzdání diplomové práce: **26. května 2023**



doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 7. prosince 2022

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 24.5.2023

v.r. Jan Sadílek
podpis studenta

ABSTRAKT

Tato práce se zaměřuje na využití jazyka Python a frameworku PyScript pro webový vývoj, který umožňuje vývoj webových aplikací přímo v prohlížeči. V teoretické části práce jsou popsány základní funkce frameworku, jeho možné implementace, porovnání s jinými podobnými frameworky a překážky, na které lze narazit. V praktické části je prezentována funkčnost tohoto frameworku skrze webovou aplikaci s jednoduchým tutoriálem, která je srozumitelná jak pro odborníky, tak pro širší veřejnost.

Klíčová slova: web, vývoj, Python, framework, PyScript, tutoriál

ABSTRACT

This work focuses on the possibilities of web development using the Python language, specifically the PyScript framework, which enables development in a web browser environment. The theoretical part describes the basic functionality, possible implementations, comparison with other similar frameworks, and its pitfalls. The practical part of the work focuses on demonstrating the functionality and presenting it to both experts and non-experts through a web application with a clear tutorial.

Keywords: web, development, Python, framework, PyScript, tutorial

Děkuji panu doktorovi Žáčkovi za skvělé rady a perfektní vedení práce.

Prohlašuji, že odevzdaná verze bakalářské/diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 WEBOVÝ VÝVOJ	11
1.1 VÝVOJ V JAZYCE C#.....	19
1.1.1 Výhody jazyka C#.....	19
1.1.2 Nevýhody jazyka C#.....	20
1.2 VÝVOJ V JAZYCE PHP	21
1.2.1 Výhody jazyka PHP	21
1.2.2 Nevýhody jazyka PHP	22
1.3 VÝVOJ V JAZYCE JAVA	22
1.3.1 Výhody jazyka Java	23
1.3.2 Nevýhody jazyka Java.....	23
1.4 VÝVOJ V JAZYCE KOTLIN	23
1.4.1 Výhody jazyka Kotlin	24
1.4.2 Nevýhody jazyka Kotlin	24
1.5 VÝVOJ V JAZYCE JAVASCRIPT.....	25
1.5.1 Výhody jazyka Javascript	26
1.5.2 Nevýhody jazyka Javascript.....	26
1.6 VÝVOJ V JAZYCE TYPESCRIPT.....	26
1.6.1 Výhody jazyka TypeScript.....	27
1.6.2 Nevýhody jazyka TypeScript.....	27
1.7 VÝVOJ V JAZYCE PYTHON	28
1.7.1 Výhody jazyka Python	28
1.7.2 Nevýhody jazyka Python	29
2 PYSCRIPT	30
2.1 TECHNOLOGIE PYSCRIPTU	31
2.1.1 WASM	31
2.1.2 Pyodide.....	33
2.1.3 Emscripten.....	35
2.1.4 CPython.....	36
3 SROVNÁNÍ PODOBNÝCH FRAMEWORKŮ	39
3.1 BRYTHON	40
3.2 TRANSCRIPT.....	41
3.3 SKULPT	42
3.4 PYWEBVIEW.....	43
3.5 PYREACT	44
3.6 ROPYDSCRIPT	45

3.7	BEEWARE	46
3.8	KIVY	47
3.9	STREAMLIT.....	48
3.10	PYJS	49
3.11	JUPYTER NOTEBOOK	50
II	PRAKTICKÁ ČÁST	52
4	POUŽITÉ TECHNOLOGIE.....	53
4.1	DJANGO.....	53
4.2	HTML5.....	54
4.3	PIP.....	56
5	PRAKTICKÉ UKÁZKY TUTORIÁLU	57
5.1	ÚVODNÍ STRANA	57
5.2	UKÁZKA 1	58
5.3	UKÁZKA 2	59
5.4	UKÁZKA 3	60
5.5	UKÁZKA 4	61
5.6	UKÁZKA 5	62
5.7	UKÁZKA 6	63
5.8	UKÁZKA 7	63
5.9	EDITOR.....	64
6	ZHODNOCENÍ FRAMEWORKU PYSRIPT.....	66
6.1.1	Soubory s příponou .whl	67
6.2	VÝZNAMNÉ BALÍČKY	67
6.2.2	Pandas	69
6.2.3	Pillow	69
6.2.4	Bokeh	70
6.2.5	Scipy.....	71
6.2.6	Scikit-learn	72
6.2.7	Beautiful Soup.....	72
6.2.8	Numpy.....	73
6.2.9	Matplotlib.....	74
	ZÁVĚR	75
	SEZNAM POUŽITÉ LITERATURY	76
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	83
	SEZNAM OBRÁZKŮ	84
	SEZNAM PŘÍLOH.....	85

ÚVOD

Webové aplikace jsou dnes nezbytnou součástí moderní společnosti a téměř každá společnost potřebuje prezentovat svou činnost na internetu, případně nabízet své produkty k prodeji. S narůstající poptávkou po interaktivních a uživatelsky přívětivých webových rozhraních se stala klíčovou potřeba efektivních a snadno použitelných frameworků. Programovací jazyk Python získal mezi vývojáři popularitu díky své jednoduchosti, univerzálnosti a silným knihovnám. V tomto kontextu se framework PyScript, založený na jazyce Python a umožňující vývoj webových aplikací přímo v prohlížeči, stal slibnou možností pro webové vývojáře. Tato diplomová práce si klade za cíl prozkoumat možnosti, které PyScript nabízí a porovnat je s jinými podobnými frameworky, kterých v současné době existuje velmi pestrá škála. Z tohoto důvodu může být někdy pro vývojáře těžké rozhodnout, zdali je vybraný framework či jazyk nejvhodnější pro specifickou implementaci. V teoretické části práce jsou popsány základní funkcionality PyScriptu a jeho využití v praxi. Zmíněn je webový vývoj nejen v jazyce Python, ale zahrnuje i obecnější pohled na vývoj webových aplikací v různých jazycích, jež se nejvíce používají. Text uvádí srovnání nejpoužívanějších jazyků pro webový vývoj a hodnotí jejich výhody a nevýhody v kontextu moderních webových aplikací. Dále jsou zde prezentovány možnosti implementace různých funkcionalit a jejich výhody a nevýhody v kontextu použití PyScriptu. V praktické části práce je demonstrována funkčnost PyScriptu na webové aplikaci s názvem "PyScript Tutoriál", která slouží jako prostředek pro přiblížení funkcionality tohoto frameworku odborníkům i neoborné veřejnosti. Součástí webové aplikace je také přehledný tutoriál, který pomáhá uživatelům se seznámením s PyScriptem a jeho použitím v praxi.

I. TEORETICKÁ ČÁST

1 WEBOVÝ VÝVOJ

Webový vývoj je proces tvorby a úpravy webových stránek a aplikací, které jsou přístupné prostřednictvím internetu. Webový vývoj zahrnuje širokou škálu dovedností a technologií, včetně návrhu webového rozhraní, vývoje backendu a frontendu, api, testování, správy databází a správu serverů. Backend část aplikací pracuje s daty a logikou definované byznysovými požadavky. Frontend se zaměřuje na přívětivé, přehledné a jasné zobrazení poskytnutých dat ze strany backendu. Neméně důležitou součástí je i API, jež slouží ke komunikaci mezi frontendem a backendem. API může být využito i pro komunikaci s ostatními externími službami. Integrace ostatních služeb zefektivňuje a zrychluje samotný vývoj. Často se takto dá vyhnout reimplementaci již vytvořeného kódu, který požadovanou funkcionalitou disponuje.

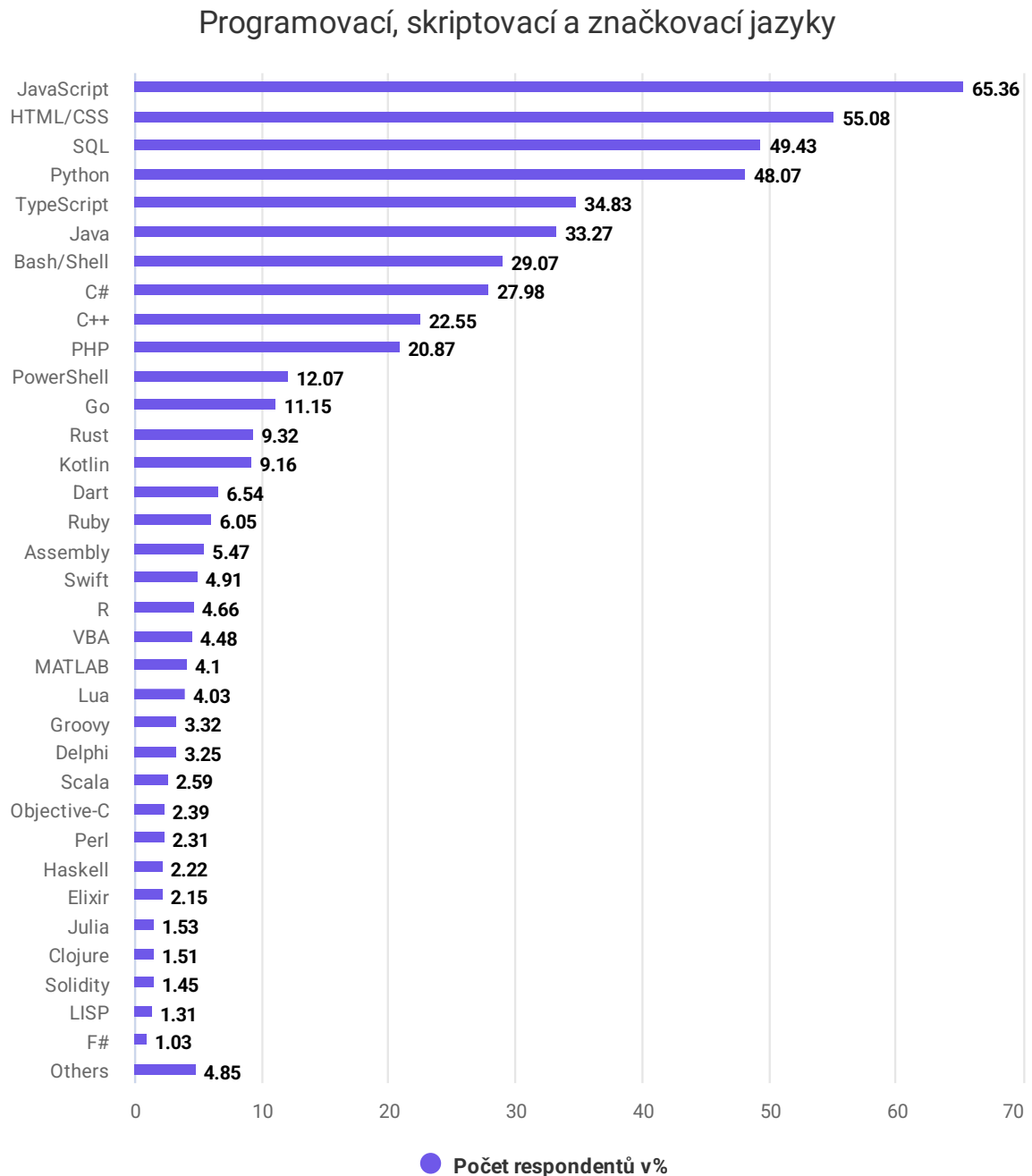
Pro zdárný vývoj webových stránek a aplikací je potřeba mít dobře definované cíle a strategie, které zahrnují pochopení potřeb uživatelů, konkurenční analýzu a výběr vhodných technologií. Další důležitou součástí je design webových stránek, kde je vhodné se zaměřit nejen na celkový vzhled, ale hlavně na uživatelskou přívětivost aplikace.

V současné době existuje mnoho nástrojů a platforem, které usnadňují webový vývoj a umožňují vytváření moderních a interaktivních webových stránek a aplikací. V následujícím oddílu jsou popsány nejčastěji používané programovací jazyky a jejich frameworky v současnosti. Pro každou z nich jsou shrnuty jejich výhody a nevýhody, krátký popis, pro jaké aplikace se nejčastěji používají a srovnání s ostatními technologiemi. Srovnávané jazyky z výběru jsou C#, PHP, JavaScript, Python, TypeScript, Java a Kotlin.

Obecně nelze jednoznačně určit nejlepší programovací jazyk. Každá aplikace má své specifické požadavky, na které mohou být vhodné pouze určité programovací jazyky. Je tedy nutné vybrat jazyk, který bude nejvhodnější pro danou aplikaci, a to na základě požadavků a charakteristik aplikace.

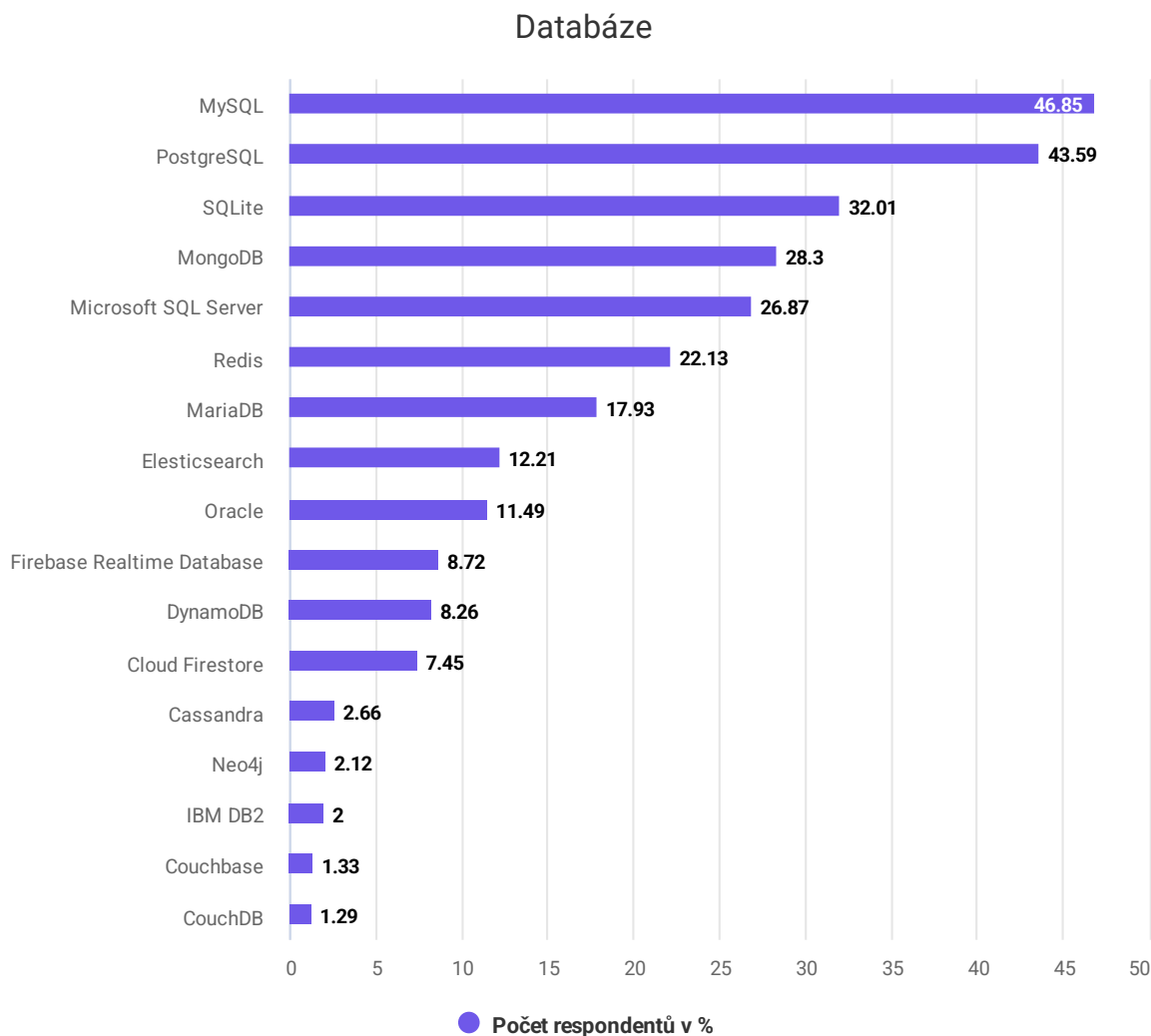
Uvedený průzkum [1] provedený na populární stránce pro vývojáře StackOverflow ilustruje realitu spojenou s vývojem softwaru. Z průzkumu bylo vybráno šest otázek nejlépe vystihující současnou situaci. Otázky jsou srozumitelně přeloženy z angličtiny, aby zachovaly svůj původní význam.

Jaké programovací, skriptovací a značkovací jazyky jste nejvíce využili pro vývoj v posledním roce a v jakém chcete pracovat v následujícím roce? (Pokud jste vybrali stejný jazyk pro obě možnosti, zaškrtněte tuto možnost.) Na tuto otázku odpovědělo 71,547 respondentů.



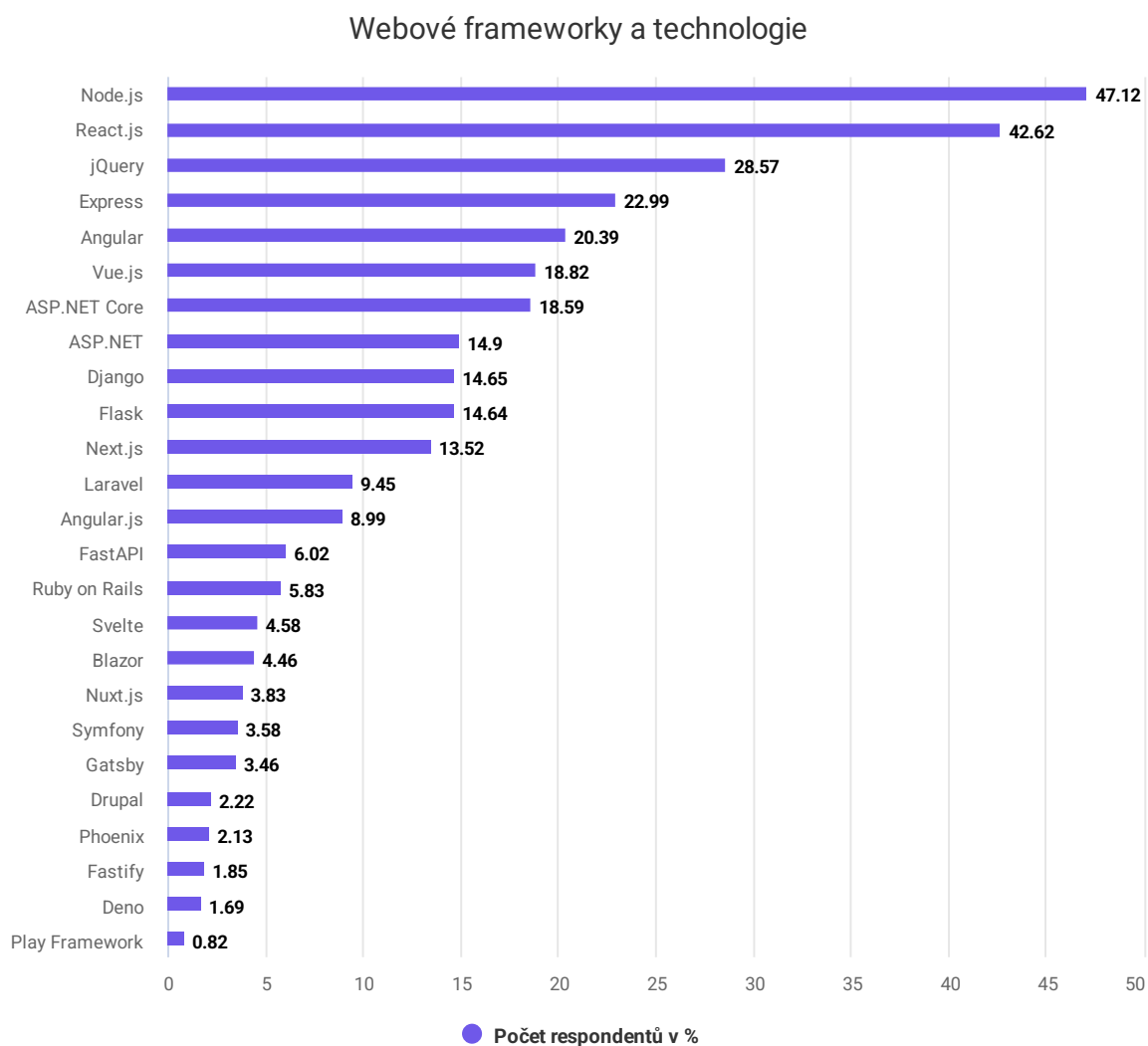
Obrázek 1 – Programovací, skriptovací a značkovací jazyky [1]

Jaké databázové prostředí jste nejvíce využili pro vývoj v posledním roce a v jakém chcete pracovat v následujícím roce? (Pokud jste vybrali stejné prostředí pro obě možnosti, zaškrtněte tuto možnost.) Na tuto otázku odpovědělo 63,327 respondentů.



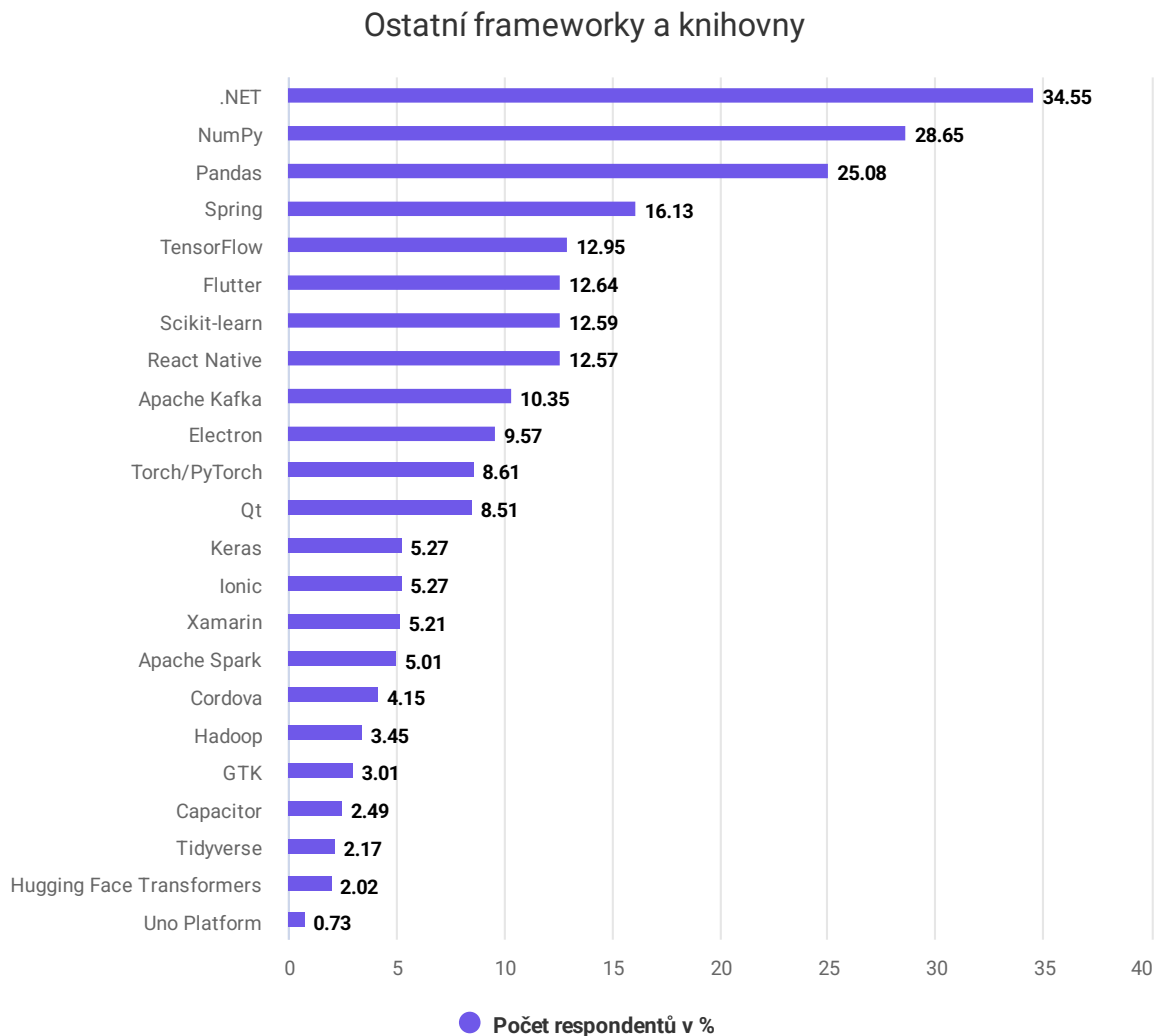
Obrázek 2 – Databáze [1]

Které webové frameworky a technologie jste nejvíce využili pro vývoj v posledním roce a v jakém chcete pracovat v následujícím roce? (Pokud jste vybrali stejné frameworky a technologie pro obě možnosti, zaškrtněte tuto možnost.) Na tuto otázku odpovědělo 58,743 respondentů.



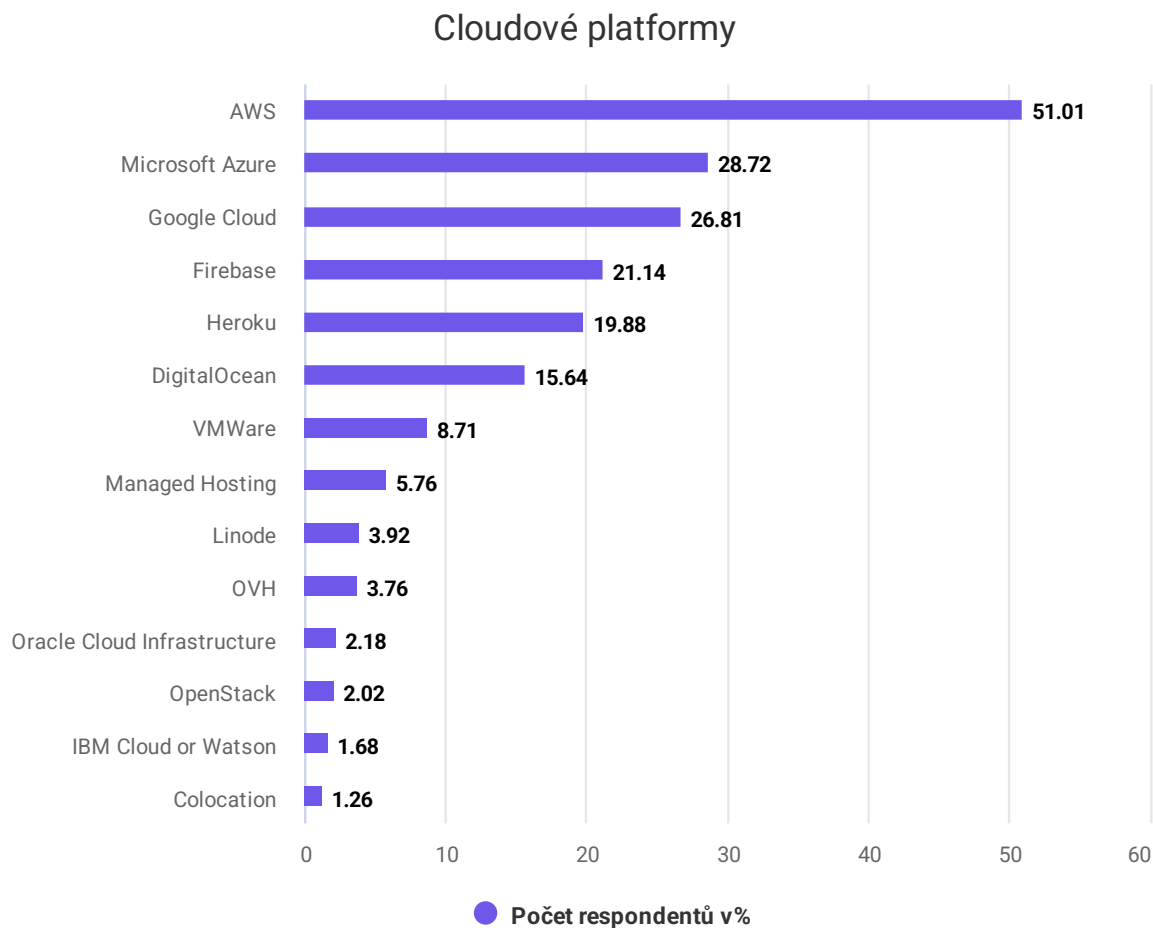
Obrázek 3 – Webové frameworky a technologie [1]

Které další frameworky a knihovny jste nejvíce využili pro vývoj v posledním roce a v jakých chcete pracovat v následujícím roce? (Pokud jste vybrali stejné vývojové prostředí pro obě možnosti, zaškrtněte tuto možnost.) Na tuto otázku odpovědělo 54,414 respondentů.



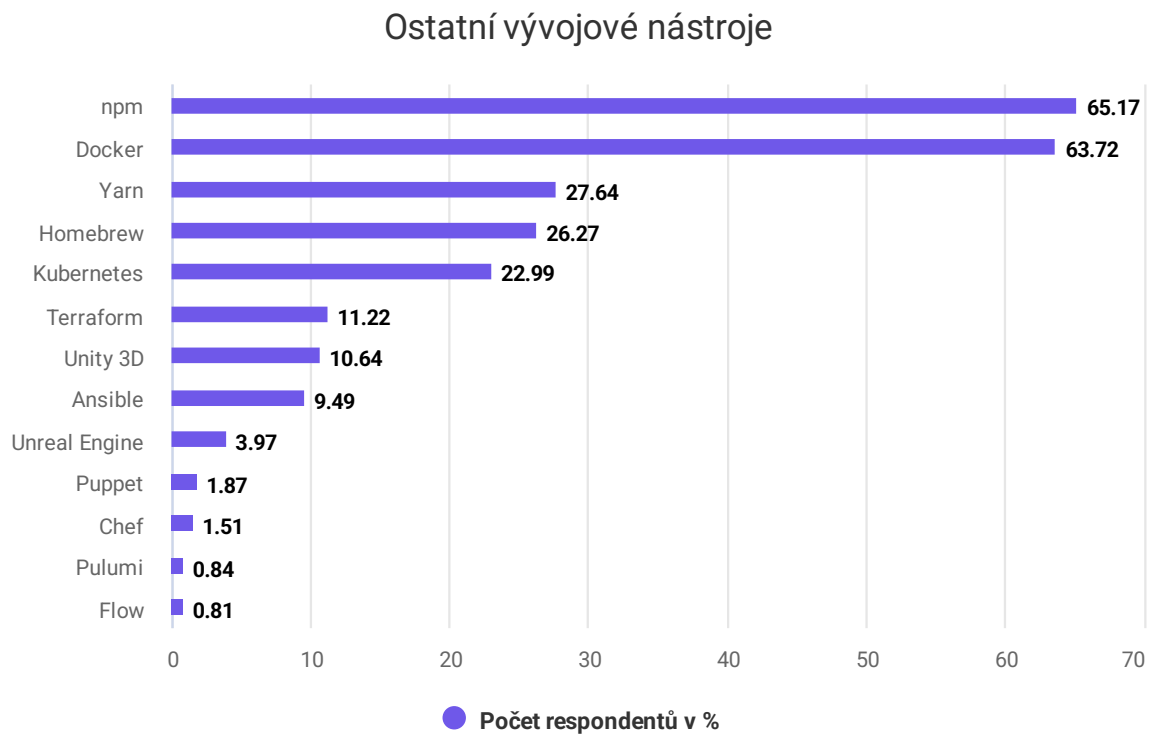
Obrázek 4 – Ostatní frameworky a knihovny [1]

Jakou cloudovou platformu jste nejvíce využili pro vývoj v posledním roce a v jakém chcete pracovat v následujícím roce? (Pokud jste vybrali stejnou platformu pro obě možnosti, zaškrtněte tuto možnost.) Na tuto otázku odpovědělo 56,553 respondentů.



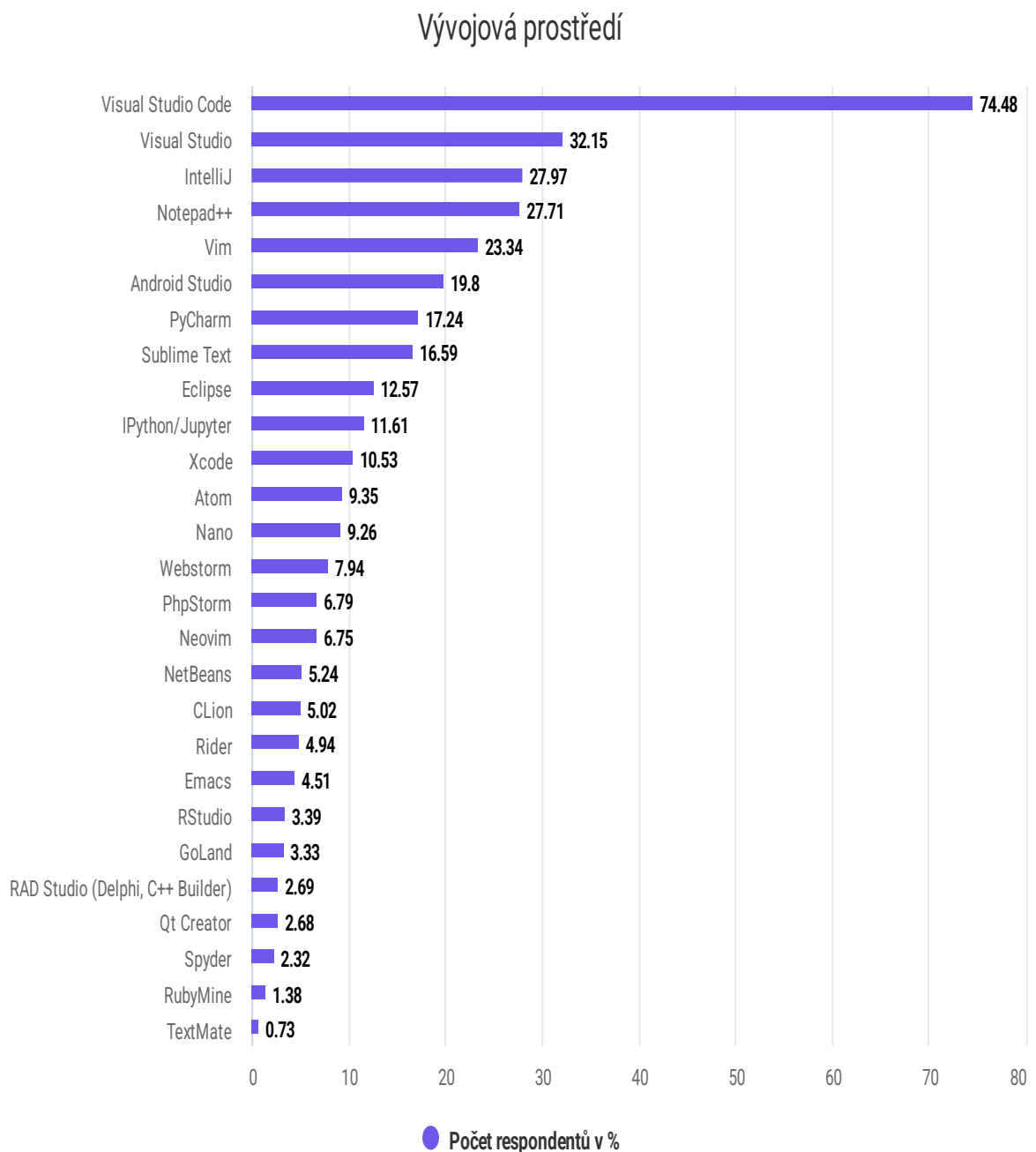
Obrázek 5 – Cloudové platformy [1]

Které vývojové nástroje jste nejvíce využili pro vývoj v posledním roce a v jakém chcete pracovat v následujícím roce? (Pokud jste vybrali stejné vývojové prostředí pro obě možnosti, zaškrtněte tuto možnost.) Na tuto otázku odpovědělo 59,919 respondentů.

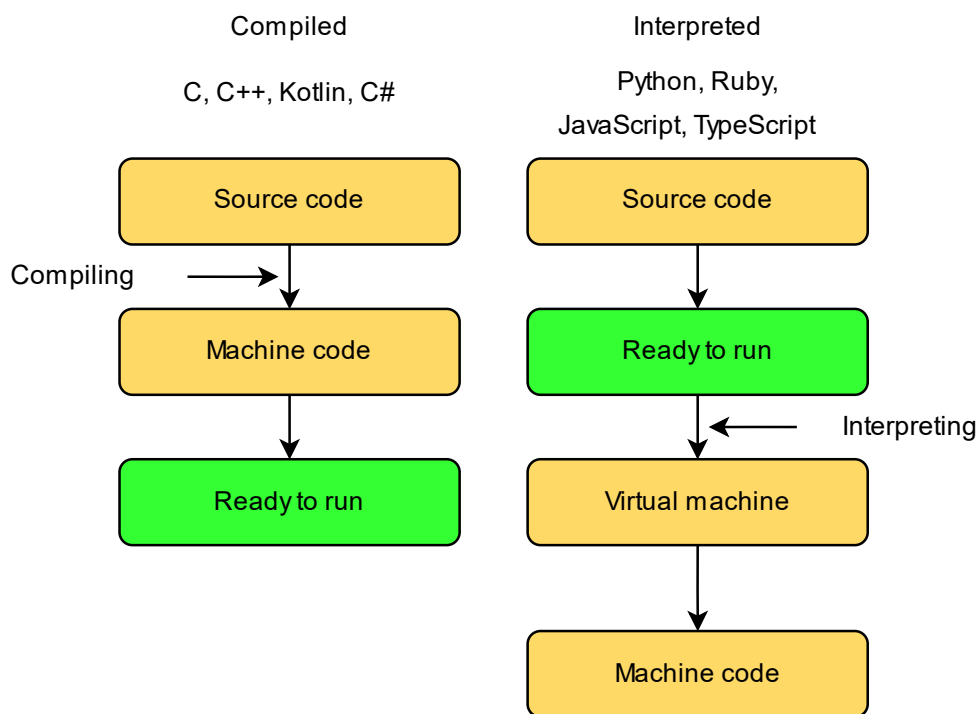


Obrázek 6 – Ostatní vývojové nástroje [1]

Která vývojová prostředí jste využili na pravidelné bázi v uplynulém roce a se kterými plánujete vyvíjet v dalším roce? Prosím zaškrtněte všechny, již se to týká. Na tuto otázku odpovědělo 71,010 respondentů.



Obrázek 7 – Vývojová prostředí [1]



Obrázek 8 – Kompilované a interpretované jazyky [2]

1.1 Vývoj v jazyce C#

Webový vývoj v jazyce C# patří mezi velmi oblíbené a rozšířené technologie pro tvorbu moderních výkonných webových aplikací. C# je objektově orientovaný programovací jazyk, který byl vyvinut společností Microsoft a používá se pro vývoj široké škály aplikací včetně desktopových, webových a mobilních aplikací.

Webový vývoj v jazyce C# se obvykle provádí pomocí frameworků ASP.NET nebo ASP.NET Core, které umožňují snadnou tvorbu webových stránek a aplikací. Tato technologie umožňuje vývojářům vytvářet dynamické a interaktivní webové aplikace s vysokou úrovní bezpečnosti a výkonu.

Webový vývoj v jazyce C# je však stále velmi populární a využívá se především pro vývoj aplikací, které vyžadují vysokou úroveň bezpečnosti a výkonu. Pokud vývojáři ovládají jazyk C# a jsou obeznámeni s frameworky ASP.NET a ASP.NET Core, mohou vytvářet kvalitní webové aplikace pro své zákazníky nebo pro své vlastní projekty. [3]

1.1.1 Výhody jazyka C#

Mezi hlavní výhody webového vývoje v jazyce C# patří:

- Široká podpora a rozšířenost – C# a frameworky ASP.NET a ASP.NET Core jsou velmi populární technologie, což znamená, že je snadné najít podporu a zdroje pro vývoj a řešení problémů.
- Vysoká bezpečnost – C# a frameworky ASP.NET poskytují vysokou úroveň bezpečnosti, včetně ochrany proti útokům na webové stránky a aplikace.
- Široká škála knihoven a nástrojů – C# a frameworky ASP.NET a ASP.NET Core mají k dispozici širokou škálu knihoven a nástrojů pro vývoj, testování a správu aplikací.
- Vysoká výkonnost – Díky kompilaci do nativního kódu C# umožňuje vývojářům vytvářet výkonné webové aplikace s minimální reží.

1.1.2 Nevýhody jazyka C#

Do nevýhod se řadí tyto položky:

- Náročnost a složitost – C# a frameworky ASP.NET a ASP.NET Core jsou poměrně náročné na naučení a použití. Z tohoto důvodu může být pro začátečníky obtížné začít rychle programovat.
- Omezená multiplatformnost – C# a frameworky ASP.NET a ASP.NET Core jsou především určené pro vývoj na platformě Windows, což může být pro některé projekty omezující.
- Závislost na technologiích Microsoftu – Webový vývoj v jazyce C# je úzce spojen s technologiemi a nástroji Microsoftu, což může být problematické.
- Cena – Některé nástroje a služby pro webový vývoj v jazyce C# mohou být relativně drahé, což může být problémem pro menší firmy nebo jednotlivé vývojáře.
- Omezená flexibilita – Někdy může být složité upravit některé funkce nebo vlastnosti frameworků ASP.NET a ASP.NET Core, což může omezovat vývojářskou flexibilitu.

I přes některé nevýhody je webový vývoj v jazyce C# stále velmi žádanou technologií, zejména pro vývoj větších a komplexnějších webových aplikací. [4] [5] [6]

1.2 Vývoj v jazyce PHP

PHP je skriptovací jazyk, který byl vytvořen pro tvorbu dynamických webových stránek. Je to interpretovaný jazyk, což znamená, že kód je přeložen a prováděn přímo na serveru, nikoli na straně klienta. To umožňuje PHP provádět složitější operace, jako jsou manipulace s daty v databázích nebo generování obsahu webových stránek, bez potřeby přenášet velké objemy dat mezi serverem a klientem. [7]

PHP poskytuje mnoho funkcí a knihoven, které usnadňují tvorbu webových stránek. Tyto funkce zahrnují práci s databázemi, manipulaci s řetězci a soubory, generování HTML, řízení toku programu a mnoho dalšího. PHP lze také použít pro tvorbu dynamických grafů a výkresů, práci s XML a JSON daty, integraci s dalšími webovými technologiemi a mnoho dalšího.

Pro práci s PHP bylo vytvořeno velké množství frameworků, mezi něž patří Laravel, CodeIgniter, CakePHP, Slim, Symfony, Zend Framework a celá řada dalších. Tyto frameworky mají za cíl usnadnit vývoj a zefektivnit tvorbu webových stránek. Nabízí celou řadu nástrojů jako jsou ORM (Object-Relational Mapping), routování, validace formulářů, cache management, testování a mnoho dalšího. Díky tomu lze dosáhnout funkční aplikace za kratší dobu s menším množstvím nutného kódu. [8] [9]

1.2.1 Výhody jazyka PHP

- Jednoduchost – PHP je snadno naučitelný a jednoduchý jazyk, což umožňuje vývojářům rychle vytvářet webové aplikace.
- Široká podpora – PHP je jedním z nejrozšířenějších jazyků pro webový vývoj a je podporován na mnoha platformách, což umožňuje snadné nasazení aplikací na různé servery.
- Široká komunita – PHP má velkou komunitu vývojářů, kteří se podílejí na jeho rozvoji a poskytují mnoho užitečných nástrojů a knihoven.
- Velké množství knihoven a frameworků – PHP nabízí mnoho knihoven a frameworků, které usnadňují vývoj webových aplikací, jako například Laravel, Symfony, nebo CodeIgniter.
- Snadná integrace s databázemi – PHP umožňuje snadnou integraci s různými databázovými systémy, což usnadňuje vývojářům práci s daty a databázemi.

- Multiplatformní jazyk – PHP poskytuje možnost využití na pestré škále platformem
- Rychlost načítání stránek – v případě pomalého internetového připojení má oproti ostatním kratší čas potřebný k načtení základní funkcionality
- Open-source – neboli otevřený software tedy PHP je vyvíjeno komunitně a kód je tak veřejně přístupný

1.2.2 Nevýhody jazyka PHP

- Bezpečnost – PHP může být zranitelný pro různé útoky, pokud není správně konfigurován a spravován.
- Pomalost – PHP může být pomalejší než některé jiné jazyky, což může zpomalit výkon webových aplikací.
- Omezení v objektovém programování – PHP má některá omezení v objektovém programování, což může být pro některé vývojáře problém.
- Složitost velkých projektů – PHP může být obtížnější použít pro vývoj velkých projektů, protože může být obtížné udržovat a spravovat velké množství kódu.
- Omezenost objektového programování – absence tříd, dědičnosti

1.3 Vývoj v jazyce Java

Java je velmi populárním jazykem pro webový vývoj, který byl vyvinut společností Sun Microsystems (nyní součástí společnosti Oracle) v roce 1995. Java je využívána na všech platformách díky své platformní nezávislosti.

Výhodou Java pro webový vývoj je široká podpora frameworků, knihoven a nástrojů, které usnadňují vývoj webových aplikací a zlepšují produktivitu vývojářů. Java také nabízí vynikající možnosti pro objektově orientované programování a podporuje mnoho návrhových vzorů.

Java má také širokou podporu a spolehlivost, což znamená, že je vhodná pro velké webové aplikace s vysokou zátěží. Java je také vhodná pro enterprise aplikace a má silnou podporu pro bezpečnost, což ji činí ideální pro finanční a bankovní aplikace.

Nevýhodou Java pro webový vývoj je, že se jedná o jazyk s vysokou náročností na výkon, což může způsobit zpomalení webových aplikací. Java také může být náročná na učení pro

začínající vývojáře, což znamená, že by mohla být složitější pro menší projekty nebo pro jednodušší webové stránky.

V každém případě je Java vynikající volbou pro vývoj webových aplikací s vysokou zátěží a pro enterprise aplikace, které vyžadují bezpečnost a spolehlivost. Využitím vhodných frameworků a nástrojů lze také výrazně usnadnit a zefektivnit vývoj webových aplikací v Javě. [14]

1.3.1 Výhody jazyka Java

- Platformě nezávislý: Java kód může být spuštěn na různých operačních systémech bez nutnosti úprav kódu, což je velkou výhodou pro vývojáře.
- Bezpečnost: Java má vynikající bezpečnostní funkce, jako je například vlastní ochrana paměti, která zabraňuje útokům typu buffer overflow, a bezpečnostní model založený na povolení, který řídí přístup k systému.
- Vysoká výkonnost: Java poskytuje vysokou výkonnost a vysokou úroveň spolehlivosti, což je klíčové pro webové aplikace, které musí být k dispozici 24/7.
- Množství knihoven a nástrojů: Java má rozsáhlou knihovnu a mnoho nástrojů, které usnadňují vývoj webových aplikací.

1.3.2 Nevýhody jazyka Java

- Složitost: Java je známý pro svou složitost, což může zpomalit vývoj a být obtížné pro začátečníky.
- Pomalejší než jiné jazyky: Java může být pomalejší než jiné jazyky, jako je například C++, a může mít vyšší nároky na systémové prostředky, což může být problém pro aplikace s vysokým počtem uživatelů.
- Paměťové nároky: Java má poměrně vysoké paměťové nároky, což může být problém pro aplikace s omezenou pamětí.

1.4 Vývoj v jazyce Kotlin

Kotlin je multiplatformní jazyk, což znamená, že lze použít stejný kód pro různé platformy, jako jsou Android, Windows, Linux a jiné pro napsání backendu a frontendu. Kotlin také nabízí mnoho moderních funkcí, jako jsou rozšířené funkce, lambdy, korutiny, null-safety a mnoho dalších. Tyto funkce pomáhají zrychlit vývoj a zlepšit kvalitu kódu. Kotlin se také

rychle rozvíjí a stává se stále populárnějším jazykem vývojářů a firem. Kotlin je staticky typovaný programovací jazyk, který běží na virtuálním stroji Java (JVM) a byl vyvinut společností JetBrains. V roce 2017 jej společnost Google oznámila jako jazyk první třídy pro vývoj v systému Android. [17]

1.4.1 Výhody jazyka Kotlin

- **Bezpečný jazyk:** Kotlin je bezpečný jazyk s pevnou typovou kontrolou a přísnými pravidly kontroly null hodnot. Tento přístup pomáhá předejít běhovým chybám a zlepšuje celkovou spolehlivost aplikace.
- **Interoperabilita s Javou:** Kotlin lze snadno použít s existujícím kódem Javy, což umožňuje postupné přecházení na Kotlin bez nutnosti náhlého přepsání celého kódu.
- **Zvýšená produktivita:** Kotlin má jednoduchou syntaxi a čitelnost, což znamená, že se kód píše rychleji a s menším množstvím chyb. Mnoho funkcí, jako např. lambdy, rozšířené funkce a výrazové funkce, také pomáhá zrychlit vývoj. V porovnání s Javou je kód Kotlinu výrazně kratší a jednodušší na pochopení.
- **Méně kódu:** Kotlin umožňuje psát méně kódu oproti Javě, což znamená, že se zlepšuje čitelnost a snižuje se riziko chyb.
- **Funkcionální prvky:** Kotlin zahrnuje funkcionální prvky, jako jsou lambda funkce, což umožňuje psát kód v moderním a efektivním stylu.

1.4.2 Nevýhody jazyka Kotlin

- **Malá komunita:** Kotlin je poměrně nový jazyk, takže komunita a podpora jsou stále menší než u Javy nebo jiných jazyků s delší historií. To může mít vliv na kvalitu knihoven a nástrojů, které jsou k dispozici pro Kotlin.
- **Výkon:** Kotlin je vyšší úrovní jazykem než Java, což může mít vliv na výkon. Nicméně výkonové rozdíly jsou obecně malé a závisí na konkrétní aplikaci. Kotlin má pomalejší kompilaci než Java, což může být nevýhodou pro velké projekty.
- **Menší podpora některých knihoven:** Zatímco Kotlin má vynikající interoperabilitu s Javou, některé knihovny nejsou plně podporovány v Kotlinu, což může být problém pro některé projekty.

Spring Boot: Jedná se o velmi populární framework pro vývoj webových aplikací v Javě a Kotlinu. Poskytuje mnoho nástrojů pro rychlý vývoj a efektivní řešení běžných problémů.

Ktor: Je novější framework speciálně navržen pro vývoj webových a serverových aplikací v Kotlinu. Jedná se o lehký a rychlý framework, který se snadno používá.

Vert.x: Je framework pro vývoj asynchronních a vícevláknových aplikací, který podporuje Kotlin a další jazyky. Poskytuje mnoho nástrojů pro vývoj reaktivních a skalovatelných aplikací.

Micronaut: Je moderní framework pro vývoj mikroslužeb v jazyce Kotlin a dalších jazycích. Jeho cílem je snížit velikost a zrychlit spouštění aplikací.

Koin: Je framework pro Dependency Injection (DI) v jazyce Kotlin, který je snadno použitelný a závisí pouze na běhovém prostředí jazyka Kotlin.

Tyto frameworky poskytují vývojářům mnoho nástrojů a funkcionalit, které usnadňují vývoj aplikací v jazyce Kotlin a zvyšují produktivitu. [18]

1.5 Vývoj v jazyce JavaScript

JavaScript je interpretovaný programovací jazyk, dále jen JS, který se používá pro tvorbu interaktivních webových stránek a webových aplikací. JS je také na straně klienta, což snižuje nároky na servery. JS je univerzální a lze jej použít pro tvorbu webových aplikací na všech platformách, včetně mobilních zařízení a desktopů.

JS se využívá pro tvorbu různých typů webových aplikací, včetně statických stránek, e-shopů, sociálních sítí a dalších. Webové aplikace se dnes často vytvářejí pomocí JS frameworků a knihoven, které usnadňují vývoj a zajišťují konzistentní vzhled a chování aplikace na různých platformách. Lze jej použít k vytváření bohatých rozhraní a vylepšení uživatelského prostředí webu

Pro vývoj webových aplikací v JS se používají nástroje jako editor kódu (např. Visual Studio Code, Atom, Sublime Text), správce balíčků (např. npm, Yarn) a nástroje pro automatizaci procesů (např. Grunt, Gulp, Webpack). Tyto nástroje usnadňují vývoj webových aplikací a zvyšují produktivitu vývojářů.

Vývoj webových aplikací v JS je velmi dynamický a neustále se vyvíjí. Každým rokem přibývají nové technologie a nástroje, které umožňují vývojářům vytvářet pokročilé webové aplikace. Populární frameworky jsou React, Angular, Vue.js, Express, and jQuery. [22]

1.5.1 Výhody jazyka Javascript

- Interaktivita: JS umožňuje vytvořit interaktivní uživatelské rozhraní, které může reagovat na uživatelské vstupy bez nutnosti komunikace se serverem.
- Univerzálnost: JS se používá na různých platformách a v různých prostředích, včetně webových prohlížečů, serverových aplikací a mobilních aplikací.
- Rychlost: Skripty v JS se běžně provádějí rychleji než jiné skriptovací jazyky, jako je například PHP.
- Široká podpora: JS má širokou podporu ze strany komunity a je k dispozici mnoho knihoven a frameworků pro usnadnění vývoje.
- Flexibilita: JS umožňuje programátorům používat mnoho různých programovacích stylů, včetně procedurálního, objektového a funkcionálního programování.

1.5.2 Nevýhody jazyka Javascript

- Závislost na prohlížeči: JS je závislý na prohlížeči a jeho funkčnost může být ovlivněna jeho verzí nebo nastavením. Některé prohlížeče mohou také blokovat provádění skriptů, což může mít negativní vliv na funkčnost aplikace.
- Bezpečnostní otázky: Skripty v jazyce JS jsou běžným cílem pro útočníky, kteří mohou využít nezabezpečených míst k vložení škodlivého kódu.
- Přetížení stránky: Při použití velkého množství skriptů na jedné stránce může docházet k přetížení stránky a zpomalení načítání.
- Rozdílné implementace: Prohlížeče mohou interpretovat jazyk JS různě, což může vést k nekompatibilitě mezi různými prohlížeči.
- Debugování: Debugování skriptů v jazyce JS může být obtížné, zejména u větších a složitě

1.6 Vývoj v jazyce TypeScript

TypeScript je ve své podstatě „nástavba“ jazyka JavaScript, což znamená, že jakýkoli platný kód v jazyce JavaScript je také platným kódem v jazyce TypeScript, dále jen TS. Za vývojem stojí stejně jako u JS společnost Microsoft. TS byl vydán v roce 2012 jako open-source projekt. Byl vytvořen k řešení problémů, které vznikají při vývoji rozsáhlých aplikací v JS.

Jedním z hlavních rozdílů mezi TS a JS je, že TS podporuje statické typování, zatímco JavaScript nikoli. To znamená, že v jazyce TS můžete při deklaraci proměnné určit její typ a překladač jazyka TS bude kontrolovat, zda jsou proměnné vždy přiřazeny hodnoty správného typu. To může pomoci zabránit určitým typům chyb. Podporuje rozhraní, která umožňují definovat podobu objektu. To může být užitečné při práci se složitými datovými strukturami. Celkově je jazyk TS vhodnější pro rozsáhlejší aplikace, zatímco jazyk JS je vhodnější pro aplikace malého rozsahu. Pro vývoj v TS kompilátor, který přeloží TS kód do běžného JS, který lze spustit v prohlížeči. [27]

1.6.1 Výhody jazyka TypeScript

- Typová kontrola: TS obsahuje silnou typovou kontrolu, což znamená, že vývojáři musí před použitím proměnných a funkcí definovat jejich typy. Toto zvyšuje bezpečnost kódu, protože kompilátor upozorní na chyby již při vývoji, než aby se objevily až v době běhu programu.
- Integrovaná podpora pro moderní funkce jazyka: TS podporuje nové funkce jazyka, které nejsou součástí JS, jako například lambda funkce a destrukurování objektů a polí.
- Zvyšuje srozumitelnost kódu: Díky typové kontrole a silnému systému objektů je kód v TS snadno čitelný a srozumitelný. To může zrychlit vývoj a zlepšit udržovatelnost kódu.
- Snadná integrace s existujícím kódem: TS je kompatibilní s kódem napsaným v jazyce JavaScript, což umožňuje postupné přecházení na TS a snadnou integraci s existujícími projekty.

1.6.2 Nevýhody jazyka TypeScript

- Naučení se nového jazyka: Přechod na TS z jazyka JavaScript vyžaduje určité množství úsilí a času na naučení se nového jazyka, jeho syntaxe a konceptů.
- Potřeba kompilace: TS musí být přeložen do kódu JS, aby mohl být spuštěn v prohlížeči. To může způsobit malé zpoždění při spouštění aplikace, zejména u větších projektů.

- Omezená podpora pro starší prohlížeče: Některé funkce TS mohou být nepodporovány v starších prohlížečích, což může způsobit problémy pro uživatele s těmito prohlížeči.
- Komplexnost: Přidání silné typové kontroly může způsobit větší komplexitu kódu a zvyšovat nároky na výkon. To může znamenat, že TS není nejlepší volbou pro jednoduché projekty.

1.7 Vývoj v jazyce Python

Python, který vyvinul Guido van Rossum a který byl vydán v roce 1991, je interpretovaný vysokoúrovňový funkcionální programovací jazyk. Python se používá při vývoji webových stránek, softwaru a systémových skriptů. Je oblíbený u mnoha programátorů a datových vědců, protože nabízí funkce, jako je podpora různých platforem, možnost běhu v interpretu a syntaxe podobná angličtině. Python také nabízí velké množství knihoven a nástrojů pro různé účely, což zvyšuje jeho flexibilitu a použitelnost. Mezi tyto knihovny patří například NumPy, Pandas, Matplotlib, TensorFlow nebo Scikit-learn, které jsou široce používány v oblasti datové analýzy, strojového učení a umělé inteligence.

Přestože má Python nepřeborné množství využití, ta nejoblíbenější se týkají tvorby webových aplikací, čtení a úprav databázových systémů, analýzy big dat a složitých matematických výpočtů. [31]

1.7.1 Výhody jazyka Python

- Snadná čitelnost kódu – Python má jednoduchou syntaxi, která umožňuje vývojářům rychleji psát a číst kód.
- Velké množství knihoven – Python má širokou paletu knihoven, které usnadňují práci v oblasti analýzy dat, strojového učení, web scrapingu a mnoho dalšího.
- Rychlá vývojová doba – díky snadné čitelnosti kódu a obrovskému množství knihoven lze v Pythonu rychle vyvíjet aplikace.
- Multiplatformní podpora – Python je multiplatformní a lze jej používat na mnoha operačních systémech, jako jsou Windows, Linux a macOS.
- Komunita – Python má velkou a aktivní komunitu, která vytváří nové knihovny, nástroje a tutoriály, což umožňuje vývojářům snadno se učit a rozvíjet své dovednosti.

1.7.2 Nevýhody jazyka Python

- Výkon – Python je interpretovaný jazyk a není tak rychlý jako kompilované jazyky jako C nebo Java. To může vést k pomalejšímu běhu programů, zejména při zpracování velkých datových souborů.
- GIL – Globální interpretovaný zámek v Pythonu (GIL) je návrhovým vzorcem, který umožňuje pouze jednomu vláknovému procesu v Pythonu vykonávat kód v jednom okamžiku. To může omezit výkon vícevláknových aplikací.
- Paměť – Python vyžaduje více paměti než některé jiné jazyky, což může být problémem, pokud má aplikace omezené zdroje.
- Nevhodný pro některé oblasti – Python může být méně vhodný pro oblasti, jako je vývoj her, tvorba nízkoúrovňových systémových aplikací a menší podpora pro mobilní aplikace a hardwarové projekty.

2 PYSCRIPT

PyScript je framework pro vývoj webových aplikací v jazyce Python, který umožňuje překlad Python kódu do JavaScriptu. Tím umožňuje běh aplikací napsaných v jazyce Python ve webovém prohlížeči. [32]

PyScript poskytuje jednoduchý a intuitivní způsob vývoje webových aplikací, kdy je využívána síla a pohodlnost jazyka Python. Jeho použití umožňuje programátorům psát kód v jednom jazyce a umožňuje snadné sdílení kódu mezi backendovou a frontendovou částí aplikace.

PyScript je technologie, která umožňuje psát kód v jazyce Python a spouštět jej nativně v prohlížeči. Poskytuje implementaci jazyka Python, která běží v prohlížeči, a rozhraní API pro přístup k DOM (Document Object Model) a dalším webovým rozhraním API. To umožňuje psát kód v jazyce Python, který komunikuje s webovou stránkou, a poskytuje způsob, jak vytvářet složité webové aplikace pomocí jazyka Python.

PyScript poskytuje odlehčenou implementaci jazyka Python se zaměřením na poskytování funkcí potřebných pro vývoj webových stránek. Poskytuje způsob, jak psát kód jazyka Python, který běží v prohlížeči, a je tak dobrou volbou pro vytváření webových aplikací, které vyžadují velkou interaktivitu a aktualizace v reálném čase.

Framework PyScript využívá jazyk Python 3.8 a vychází z konceptu Reactu. PyScript umožňuje programátorům vytvářet webové aplikace pomocí definování komponentů, které jsou následně složeny do stromu a vykresleny do prohlížeče. Komponenty jsou napsány v jazyce Python, přičemž jsou přeloženy do JavaScriptu, který je spuštěn v prohlížeči.

PyScript poskytuje mnoho užitečných funkcí pro vývoj webových aplikací, jako jsou funkce pro práci s událostmi, pro práci s datovými strukturami a pro práci s asynchronním programováním. PyScript také umožňuje snadnou integraci s jinými balíčky pro Python, jako jsou například Pandas, NumPy nebo Matplotlib.

Jako výhody použití frameworku PyScript lze uvést:

- Použití jednoho jazyka (Python) pro backend i frontend.
- Snadná integrace s jinými balíčky pro Python.
- Jednoduchá syntaxe pro tvorbu komponentů.
- Podpora asynchronního programování.

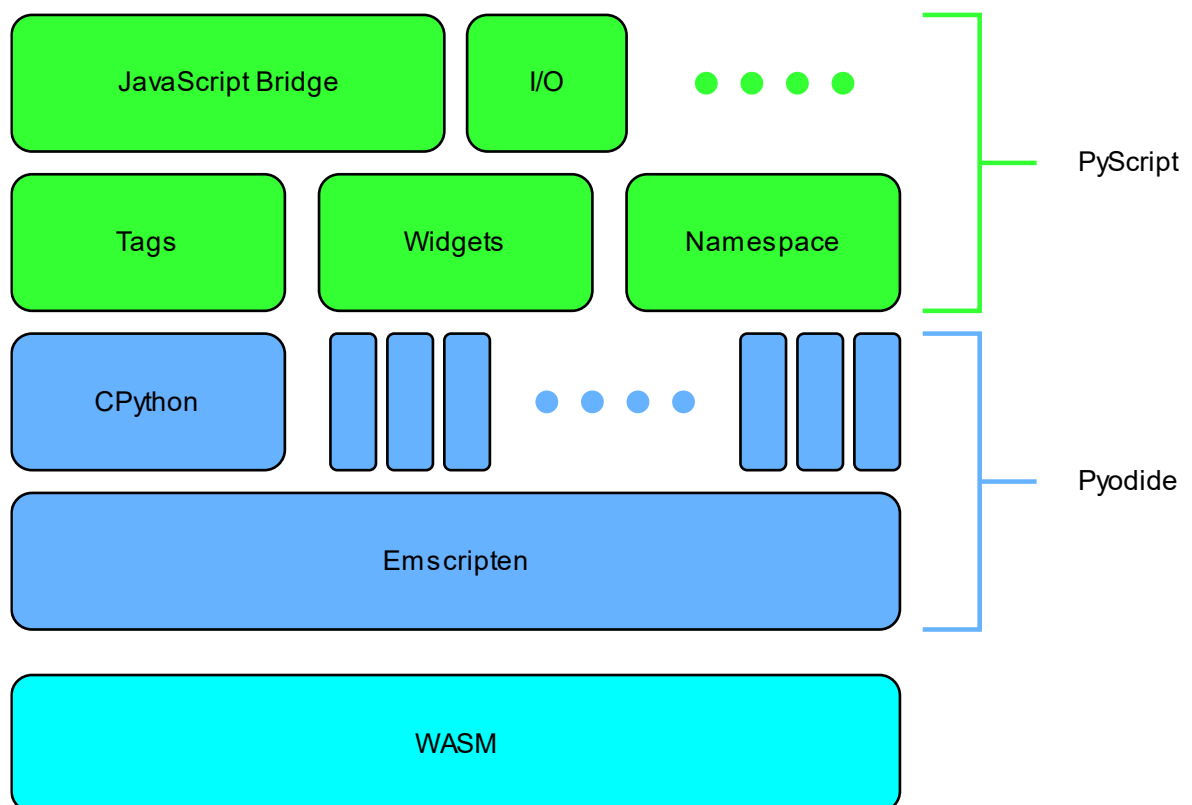
Nevýhodou použití PyScript může být:

- Omezený ekosystém a zatím vývoj stojí téměř na počátku.
- Malá komunita v porovnání s jinými webovými frameworky vzhledem k relativně nedávnému vydání.

Tvůrci sami na svých stránkách sami upozorňují, že se jedná o rané fáze vývoje. Může se tedy stát, že se objeví neočekávané chyby. Funkcionalita byla plně otestována zatím pouze v prostředí webového prohlížeče Chrome. [33] [34]

2.1 Technologie PyScriptu

PyScript stojí na Pyodide, což je port CPythonu do WebAssembly/Emscripten. Na obrázku je ilustrována základní architektura.



Obrázek 9 – Technologický přehled [40]

2.1.1 WASM

WebAssembly (Wasm) je binární formát instrukční sady určený pro webové prostředí, který umožňuje spouštění vysokoúrovňových jazyků, jako je C, C++, Rust a další, přímo v prohlížeči. Jeho hlavním cílem je umožnit vysoce výkonné webové aplikace a rozšířit možnosti vývojářů pro tvorbu sofistikovaných webových prostředí.

Zde je pár klíčových bodů o WebAssembly:

- **Výkon:** WebAssembly byl navržen s ohledem na vysoký výkon. Jeho binární formát je optimalizován pro rychlé načítání a spuštění kódu. Díky tomu může být WebAssembly mnohem efektivnější než kód napsaný v jazyce JavaScript, což umožňuje vývojářům přenášet a spouštět náročné výpočty a aplikace ve webovém prostředí.
- **Platformní nezávislost:** WebAssembly je nezávislý na platformě a prohlížeči. To znamená, že kód vytvořený v jednom prohlížeči může být spuštěn ve všech prohlížečích, které podporují WebAssembly. Tímto způsobem umožňuje vývojářům psát kód jednou a spouštět ho na různých platformách bez potřeby přepisování.
- **Interoperabilita s JavaScriptem:** WebAssembly je navržen tak, aby se integroval se stávajícím ekosystémem JavaScriptu. To znamená, že můžete volat funkce napsané v jazyce JavaScript z kódu WebAssembly a naopak. Tato interoperabilita poskytuje vývojářům flexibilitu při práci s WebAssembly a umožňuje postupnou migraci existujících aplikací.
- **Bezpečnost:** WebAssembly je navržen s důrazem na bezpečnost. Kód WebAssembly je spuštěn v omezeném prostoru, který je izolován od zbytku webové stránky, aby se minimalizovalo riziko zneužití. Tím se zajišťuje, že kód WebAssembly nemá přístup ke citlivým informacím a může být spuštěn bezpečně ve webovém prostředí.
- **Ekosystém:** WebAssembly má bohatý ekosystém nástrojů a knihoven, které usnadňují vývojářům práci s touto technologií. Existuje mnoho kompilátorů, jako je Emscripten, které umožňují překlad různých jazyků do formátu WebAssembly. K dispozici jsou také balíčkovací nástroje, které usnadňují správu závislostí a distribuci aplikací postavených na WebAssembly.

WebAssembly má široké spektrum možností použití. Mezi ně patří:

- **Portace existujících aplikací:** WebAssembly umožňuje přenést existující aplikace napsané v jiných jazycích na web bez nutnosti přepsat je do jazyka JavaScript. To usnadňuje přístup k rozsáhlému kódu a knihovnám, které již existují v těchto jazycích.

- Vysokourovňové webové aplikace: WebAssembly umožňuje vytvářet sofistikované webové aplikace, které vyžadují vysoký výkon, například hry, 3D vizualizace, video a zvukové editory nebo nástroje pro strojové učení.
- Rychlé výpočty a analýzy: WebAssembly je vhodný pro rychlé výpočty a analýzy, protože poskytuje přímý přístup k hardwarovým prostředkům prohlížeče. To je užitečné pro složité matematické výpočty, simulace, grafické a zvukové zpracování a další úlohy, které vyžadují výpočetní sílu.
- Multiplatformní aplikace: WebAssembly umožňuje vývojářům psát aplikace, které mohou být spuštěny na různých platformách, včetně desktopu, mobilních zařízení a vestavěných systémů. To znamená, že lze vytvářet aplikace, které jsou kompatibilní s různými zařízeními a operačními systémy.

WebAssembly je inovativní technologie, která přináší výkonné a efektivní možnosti pro spouštění kódu ve webovém prostředí. S jeho rychlostí, bezpečností a platformní nezávislostí poskytuje vývojářům nové možnosti pro tvorbu vysokoúrovňových webových aplikací a přenášení existujícího kódu na web. S rozšiřujícím se ekosystémem nástrojů a knihoven je WebAssembly stále populárnější volbou pro moderní webový vývoj. [41] [42]

2.1.2 Pyodide

Pyodide je open-source projekt vyvinutý společností Mozilla, který umožňuje spouštění interpretu jazyka Python v prohlížeči pomocí technologie WebAssembly. Jeho hlavním cílem je přinést plnohodnotné prostředí Pythonu do webového prostředí, což umožňuje vývojářům psát a spouštět Pythonový kód přímo ve webovém prohlížeči. [44]

Několik klíčových aspektů Pyodide:

- Integrace Pythonu a WebAssembly: Pyodide kombinuje interpret Pythonu s technologií WebAssembly, což umožňuje přenést Pythonový kód do webového prostředí. Díky této integraci mohou vývojáři využívat plnohodnotnou sílu Pythonu, jeho rozsáhlou knihovnu a ekosystém balíčků přímo ve webovém prostředí.
- Plnohodnotné prostředí Pythonu: Pyodide poskytuje vývojářům plně funkční Pythonové prostředí v prohlížeči. To znamená, že můžete psát, spouštět a ladit Pythonový kód přímo ve webovém prohlížeči bez nutnosti instalace nebo konfigurace interpreta Pythonu na straně klienta.

- Podpora pro knihovny a balíčky: Pyodide poskytuje podporu pro mnoho populárních Pythonových knihoven a balíčků, včetně numpy, pandas, matplotlib, scikit-learn a mnoha dalších. To umožňuje vývojářům využívat tyto knihovny pro výpočty, vizualizace dat a další úlohy přímo ve webovém prostředí.
- Výkon a rychlost: Díky použití technologie WebAssembly může Pyodide dosáhnout vysokého výkonu a rychlosti provádění Pythonového kódu ve webovém prostředí. WebAssembly umožňuje efektivní kompilaci a spouštění kódu, což znamená, že aplikace postavené na Pyodide mohou být výkonné a responzivní.
- Interoperabilita s JavaScriptem: Pyodide je navržen tak, aby se dobře integroval s existujícími webovými technologiemi a jazykem JavaScript. To znamená, že můžete volat Pythonový kód z JavaScriptu a naopak. Tato interoperabilita umožňuje vývojářům kombinovat sílu Pythonu s možnostmi JavaScriptu pro tvorbu pokročilých webových aplikací.

Pyodide poskytuje vývojářům řadu výhod a přínosů:

- Vzdělávání a učení: Pyodide je užitečným nástrojem pro výuku a učení programování v jazyce Python. S možností spouštění Pythonového kódu přímo ve webovém prohlížeči se studenti nemusí zabývat instalací a konfigurací interpreta Pythonu na svých počítačích. To zjednodušuje proces a umožňuje studentům rychle začít se programováním.
- Data Science a vizualizace dat: Pyodide poskytuje přístup k populárním knihovnám pro analýzu dat, jako je numpy a pandas, a knihovnám pro vizualizaci dat, jako je matplotlib. To umožňuje vývojářům provádět analýzu dat a vizualizovat výsledky přímo ve webovém prostředí, což je užitečné pro vývoj datově orientovaných aplikací a interaktivních vizualizací.
- Prototypování a testování: Pyodide je vhodným nástrojem pro rychlé prototypování a testování kódu. Díky možnosti okamžitého spouštění Pythonového kódu ve webovém prohlížeči můžete rychle vytvářet a testovat různé funkce a algoritmy, aniž byste museli kompilovat nebo spouštět kód mimo prohlížeč.
- Komunita a podpora: Pyodide je součástí širšího ekosystému nástrojů a knihoven společnosti Mozilla. Existuje aktivní komunita vývojářů, která poskytuje podporu, příspěvky a aktualizace pro Pyodide. To znamená, že můžete najít zdroje,

dokumentaci a příklady kódu, které vám pomohou při vývoji aplikací s využitím Pyodide.

Pyodide je zajímavým projektem, který umožňuje spouštění jazyka Python přímo ve webovém prohlížeči. S jeho výkonem, podporou knihoven a integrací s existujícími webovými technologiemi se stává užitečným nástrojem pro vývojáře, kteří chtějí využívat sílu Pythonu ve webovém prostředí. [45]

2.1.3 Emscripten

Emscripten je nástrojový soubor a překladač vyvinutý společností Mozilla, který umožňuje překlad C/C++ kódu do jazyka JavaScript. Jeho hlavním cílem je umožnit vývojářům přenášet existující kód napsaný v jiných jazycích, jako je C nebo C++, a spouštět ho přímo v prohlížeči pomocí webového standardu WebAssembly (Wasm). [46]

Emscripten kombinuje několik technologií, včetně LLVM kompilátoru a samotného překladače Emscripten. LLVM kompilátor převádí C/C++ kód do mezikódu, který se nazývá "bitcode". Tento bitcode je následně přeložen do jazyka JavaScript pomocí Emscripten překladače. Výsledkem je soubor JavaScript, který obsahuje přeložený kód spolu s vygenerovanými funkcemi pro komunikaci s prostředím prohlížeče.

Emscripten poskytuje vývojářům mnoho funkcí a nástrojů, které usnadňují práci s překládáním a spouštěním C/C++ kódu ve webovém prostředí. Mezi tyto funkce patří:

1. Emterpreter: Tato funkce umožňuje spouštění C/C++ kódu ve webovém prostředí postupně, místo toho, aby se kód spustil najednou. To znamená, že můžete mít více asynchronních operací nebo iterativních výpočtů, které jsou synchronizovány s webovým prostředím.
2. Podpora pro knihovny: Emscripten nabízí podporu pro mnoho populárních C/C++ knihoven, což umožňuje snadnou integraci stávajícího kódu do webových aplikací. Mezi podporované knihovny patří například SDL, OpenGL, zlib a mnoho dalších.
3. Optimalizace kódu: Emscripten provádí různé optimalizace kódu, aby dosáhl co nejlepšího výkonu ve webovém prostředí. To zahrnuje například minimalizaci velikosti výsledného JavaScript kódu a optimalizace rychlosti provádění.
4. Nástroje pro ladění: Emscripten poskytuje nástroje pro ladění a profilování kódu, což umožňuje vývojářům snadno odhalit a opravit chyby a optimalizovat výkon aplikací.

Použití Emscriptenu může být pro různé účely a scénáře. Některé z hlavních případů použití Emscripten zahrnují:

1. Přenos existujícího kódu na web: Emscripten umožňuje přenést existující C/C++ kód na web bez nutnosti přepisovat ho do jazyka JavaScript. To je užitečné pro přenášení složitých výpočtů, simulací, her a dalších aplikací, které jsou napsány v C/C++.
2. Webové rozhraní pro knihovny: Emscripten poskytuje prostředky pro překlad C/C++ knihoven do jazyka JavaScript, čímž umožňuje vývojářům vytvářet webové rozhraní pro tyto knihovny. To znamená, že můžete využívat funkcionality rozsáhlých knihoven ve webovém prostředí.
3. Hry a grafika: Emscripten má podporu pro knihovny jako SDL nebo OpenGL, což usnadňuje tvorbu webových her a aplikací s grafickým rozhraním. Můžete přenášet existující hry nebo vytvářet nové hry pomocí C/C++ a spouštět je přímo v prohlížeči.
4. Experimentování a vzdělávání: Emscripten poskytuje vývojářům možnost experimentovat s C/C++ kódem ve webovém prostředí a rychle vidět výsledky své práce. To je užitečné pro vzdělávací účely a různé prototypování.

Emscripten je mocný nástroj, který umožňuje přenášet a spouštět C/C++ kód přímo v prohlížeči pomocí technologie WebAssembly. Poskytuje vývojářům mnoho funkcí a nástrojů, které usnadňují práci s překládáním, optimalizací a laděním kódu. Pokud máte existující C/C++ kód nebo plánujete vyvíjet webové aplikace s vysokým výkonem, Emscripten by mohl být užitečným nástrojem pro vaše potřeby. [47]

2.1.4 CPython

CPython je standardní implementace jazyka Python. Byl vyvinut v jazyce C a je považován za referenční implementaci Pythonu. Název "CPython" se používá k rozlišení této implementace od jiných, jako je například Jython (implementace v jazyce Java) nebo IronPython (implementace pro platformu .NET). [48]

Zde je několik klíčových bodů o CPython:

- Interpretovaný jazyk: CPython interpretuje Python kód řádek po řádce za běhu. Interpretace znamená, že kód je přeložen do mezikódu, který je následně vykonáván interpretem. Tento přístup umožňuje rychlé prototypování a snadnou čitelnost kódu, což je jedna z charakteristik Pythonu.

- **Dynamická typovost:** Python je dynamicky typovaný jazyk, což znamená, že typy proměnných jsou určovány až za běhu programu. CPython je navržen tak, aby byl schopen efektivně pracovat s dynamickou typovostí Pythonu a zajišťovat dynamickou alokaci paměti pro proměnné.
- **Široká knihovna a kompatibilita:** CPython je dodáván s rozsáhlou knihovnou standardních modulů, které pokrývají širokou škálu funkcionalit. Tyto moduly zahrnují manipulaci s řetězci, vstup a výstup, práci s soubory, síťovou komunikaci, správu procesů a mnoho dalšího. CPython také poskytuje podporu pro rozšiřující moduly napsané v jiných jazycích, jako je C.
- **Globální interpret:** CPython poskytuje globální interpret pro Python, což znamená, že kód napsaný v CPythonu je spustitelný na různých operačních systémech a platformách. To zajišťuje vysokou přenositelnost Pythonových aplikací a umožňuje vývojářům psát kód jednou a spouštět ho na různých zařízeních.
- **Komunita a podpora:** CPython má silnou a aktivní komunitu vývojářů, kteří poskytují podporu, příspěvky a aktualizace pro tuto implementaci. Existuje také rozsáhlá dokumentace, knihy, tutoriály a online zdroje, které pomáhají vývojářům s porozuměním a využitím CPythonu.
- **Rozšiřitelnost:** CPython je navržen tak, aby byl snadno rozšiřitelný. Vývojáři mohou psát rozšiřující moduly pro CPython v jazyce C. Tato možnost umožňuje vývojářům optimalizovat kritické části kódu a využít externí knihovny napsané v jiných jazycích. Rozšiřující moduly přímo komunikují s interpretem CPythonu a umožňují manipulaci s Pythonovými objekty a vykonávání Pythonového kódu.
- **Garbage collection:** CPython používá garbage collection pro správu paměti. Garbage collector sleduje objekty v paměti a odstraňuje nepoužívané objekty, aby uvolnil paměť. Tím snižuje zátěž na vývojáře, který se nemusí starat o ruční správu paměti.
- **Výkon:** CPython je výkonnou implementací Pythonu, která byla optimalizována pro běh běžných Pythonových programů. Nicméně, kvůli dynamické typovosti a dalším vlastnostem Pythonu, CPython nemusí být nejrychlejší implementací Pythonu. Existují alternativní implementace, jako je PyPy nebo Jython, které se zaměřují na vylepšení výkonu.

- Další vlastnosti: CPython poskytuje další užitečné vlastnosti, jako je interaktivní režim, který umožňuje spouštět příkazy a kód přímo z příkazové řádky, a debugger pro ladění Pythonových programů. Také podporuje vícevláknové programování a poskytuje nástroje pro správu vláken a synchronizaci.

CPython je nejrozšířenější a referenční implementací jazyka Python. Svojí stabilitou, rozsáhlou knihovnou, komunitou vývojářů a širokou podporou se stal standardem pro vývoj Pythonových aplikací. Vývojáři využívají CPython pro psaní různorodých aplikací, od jednoduchých skriptů po složité webové aplikace a vědecké výpočty. [49]

3 SROVNÁNÍ PODOBNÝCH FRAMEWORKŮ

Frameworky podobné PyScriptu nabízejí vývojářům několik výhod a funkcionalit, které usnadňují vývoj webových aplikací v jazyce Python. Zde je obecný přehled toho, co tyto frameworky mohou poskytovat:

- Možnost psát webové aplikace v jazyce Python: To znamená, že mohou využívat své znalosti a dovednosti v Pythonu při vývoji webových projektů.
- Překlad Python kódu do JS: To umožňuje spouštět Python kód na straně klienta a poskytuje kompatibilitu s webovými technologiemi.
- Podpora interaktivního uživatelského rozhraní: Nabízejí nástroje a knihovny pro vytváření interaktivního uživatelského rozhraní ve webových aplikacích. To zahrnuje manipulaci s DOMem (Document Object Model), událostmi, animacemi a dalšími interakcemi mezi uživatelem a aplikací.
- Kompatibilita s existujícím kódem: Často se snaží být kompatibilní s existujícím kódem napsaným v jazyce Python. To znamená, že vývojáři mohou snadno migrovat své stávající aplikace nebo knihovny do webového prostředí a znovu využít existující kód.
- Optimalizovaný a efektivní výstupní kód: Generování optimalizovaného JS kódu ze zdrojového kódu napsaného v Pythonu. To zajišťuje vysokou výkonnost webových aplikací a minimalizuje jejich velikost, což je důležité pro rychlé načítání stránek.
- Ekosystém knihoven a nástrojů: Disponují rozsáhlým ekosystémem knihoven a nástrojů, které vývojářům usnadňují práci. Tyto knihovny mohou obsahovat různé funkcionality, jako je manipulace s daty, správa stavu aplikace, routování, testování a další.

Frameworky jako PyScript a podobné nabízejí vývojářům především pohodlné a efektivní prostředí pro vývoj webových aplikací v jazyce Python. Zároveň umožňují spouštění kódu napsaného v jazyce Python ve webovém prohlížeči, což otevírá nové možnosti pro tvorbu interaktivních a dynamických webových aplikací. Kromě toho, díky kompatibilitě s existujícím kódem a širokému ekosystému knihoven a nástrojů, poskytují tyto frameworky vývojářům potřebné nástroje a prostředky pro rychlý a efektivní vývoj webových aplikací.

3.1 Brython

Brython je webový framework pro vytváření interaktivních webových stránek a aplikací pomocí jazyka Python. Jeho název je zkratkou pro "browser Python" a to ukazuje na to, že Brython umožňuje spouštění kódu Pythonu přímo v prohlížeči. [50]

Brython byl vytvořen jako open-source projekt a je k dispozici zdarma pro všechny, kteří chtějí používat Python k vývoji webových aplikací. Jeho cílem je poskytnout vývojářům jednoduché, intuitivní a efektivní řešení pro tvorbu moderních webových aplikací.

Jednou z hlavních vlastností Brythonu je jeho schopnost přeložit Python kód na JavaScript kód, který může být spuštěn v prohlížeči. To znamená, že vývojáři mohou psát kód Pythonu, který bude spuštěn přímo v prohlížeči a bude interagovat s uživatelským rozhraním.

Brython také obsahuje mnoho vestavěných knihoven, které usnadňují vývoj webových aplikací. Například modul "browser" poskytuje přístup k DOM (Document Object Model) pro manipulaci s HTML a CSS, zatímco modul "ajax" umožňuje webovým aplikacím komunikovat se serverem bez nutnosti načtení celé stránky.

Kromě toho Brython také poskytuje nástroje pro tvorbu grafického uživatelského rozhraní, jako jsou například widgety pro vstup textu, tlačítka, seznamy a další. Tyto nástroje usnadňují vývojářům tvorbu interaktivních a responzivních webových stránek a aplikací.

Vývojáři mohou také použít Brython k integraci s dalšími webovými technologiemi, jako je například HTML5 Canvas pro kreslení a animaci, nebo WebGL pro vykreslování 3D grafiky.

Další výhodou Brythonu je jeho jednoduchá instalace a použití. Stačí stáhnout zdrojové soubory nebo nainstalovat balíček pomocí správce balíčků jako je pip. Poté je možné začít psát webové aplikace v Pythonu přímo v prohlížeči. Tím se vývojářům umožňuje rychlejší prototypování a testování svých nápadů.

Brython také nabízí kompatibilitu s existujícími knihovnamy a frameworky Pythonu. Mnoho populárních knihoven, napsaných v čistém Pythonu může být použito i s Brythonem. To umožňuje vývojářům využívat své znalosti a dovednosti získané při práci s Pythonem i při vývoji webových aplikací.

Je třeba poznamenat, že Brython není určen pro vývoj velkých a komplexních webových aplikací. Jeho hlavním zaměřením je snadná integrace Pythonu do webových stránek a menších aplikací. Pokud potřebujete vytvořit rozsáhlou webovou aplikaci s vysokou

výkoností, mohou být lepší volbou jiné webové frameworky, které jsou optimalizovány pro tyto účely.

Celkově lze říci, že Brython poskytuje vývojářům Pythonu jednoduché a efektivní řešení pro tvorbu moderních webových aplikací. Jeho schopnost přeložit Python kód na JavaScript umožňuje vývojářům psát kód Pythonu, který může být spuštěn přímo v prohlížeči a interagovat s uživatelským rozhraním. Brython také poskytuje mnoho vestavěných knihoven a nástrojů, které usnadňují vývoj webových aplikací. Například Brython obsahuje knihovny pro manipulaci s DOM (Document Object Model), práci s událostmi, asynchronní komunikaci se serverem a další. [51]

3.2 Transcrypt

Transcrypt je dalším zajímavým webovým frameworkem, který umožňuje vývojářům psát kód Pythonu a přeložit ho na JavaScript. Jeho cílem je poskytnout Python vývojářům možnost využívat jejich znalosti a dovednosti při tvorbě webových aplikací. [52]

Primárně je určen pro menší a středně velké webové aplikace. Pro větší a komplexnější projekty mohou být vhodnější jiné webové frameworky nebo kombinace Pythonu a JavaScriptu.

Jednou z jeho klíčových vlastností je schopnost překladu Python kódu na čistý, optimalizovaný JavaScript. To znamená, že výsledný JavaScript kód je malý, efektivní a optimalizovaný pro běh v prohlížeči. Díky tomu může být kód napsaný v Transcryptu spuštěn rychle a efektivně.

Podporuje mnoho funkcí a knihoven Pythonu, včetně těch základních, jako jsou řetězce, seznamy, slovníky a podmínky. To umožňuje vývojářům používat mnoho standardních pythonových konstrukcí při tvorbě webových aplikací. Dále podporuje objektově orientované programování (OOP) a moduly, což umožňuje organizovat kód do logických a znovupoužitelných bloků.

Další výhodou je, že nabízí snadnou integraci s existujícími JavaScript knihovnami a frameworky. Můžete tedy používat populární knihovny, jako je například jQuery nebo React, a psát kód v Pythonu, který s nimi interaguje. To umožňuje vývojářům využívat širokou škálu existujících nástrojů a komunitu spojenou s JavaScriptem.

Poskytuje několik dalších funkcí, které usnadňují vývoj webových aplikací. Například obsahuje nástroje pro práci s DOM (Document Object Model) pro manipulaci s HTML a

CSS, podporuje události a asynchronní operace a nabízí nástroje pro ladění a profilování kódu.

Nejedná se o kompletní náhradu za JavaScript. I když umožňuje psát kód Pythonu, musíte stále mít alespoň základní znalosti JS, zejména pokud chcete integrovat s existujícími JS knihovny. Transcrypt také nenabízí plnou podporu všech funkcí Pythonu, a proto mohou být některé pokročilé funkce z Pythonu a knihovny omezené nebo nedostupné.

Vyznačuje se snadnou instalací a použitím. Stačí nainstalovat jeho balíček pomocí správce balíčků jako je pip a můžete začít psát kód Pythonu pro webové aplikace. Poskytuje také nástroje pro sestavení, spouštění, ladění a optimalizaci kódu, což usnadňuje vývoj a údržbu aplikací.

Celkově lze říci, že se jedná o zajímavým nástrojem pro vývoj webových aplikací pomocí Pythonu. Jeho překlad Pythonu na optimalizovaný JS umožňuje vývojářům využívat sílu Pythonu a zároveň těžit z ekosystému JS. [53]

3.3 Skulpt

Skulpt je webovým frameworkem pro interpretaci kódu napsaného v jazyce Python přímo v prohlížeči. Jeho hlavním cílem je umožnit vývojářům používat Python v webovém prostředí a poskytnout jim prostředky k vytváření interaktivních webových stránek a aplikací.

Jednou z hlavních vlastností Skulptu je jeho schopnost přeložit Python kód na JavaScript. To znamená, že vývojáři mohou psát kód Pythonu a spouštět ho přímo v prohlížeči bez nutnosti jakékoli kompilace nebo instalace dalších nástrojů. Skulpt překládá Pythonový kód na JavaScript on-the-fly a spouští ho v prohlížeči pomocí nástrojů JavaScriptového interpretu.

Skulpt nabízí podporu většiny funkcí a knihoven Pythonu, včetně základních datových typů, podmínek, smyček, funkcí a objektově orientovaného programování. To umožňuje vývojářům psát kód v Pythonu podobným způsobem jako při vývoji desktopových aplikací. Skulpt také poskytuje některé specifické knihovny pro práci s webovým prostředím, jako je manipulace s DOM (Document Object Model), práce s událostmi a AJAX komunikace se serverem.

Další výhodou Skulptu je jeho jednoduchá integrace s existujícími webovými technologiemi. Vývojáři mohou používat HTML a CSS pro tvorbu uživatelského rozhraní a

propojovat ho s Pythonovým kódem pomocí Skulptu. Umožňuje také využívat externí JS knihovny a frameworky, což rozšiřuje možnosti vývoje webových aplikací.

Přináší také výhody sdílení kódu mezi backendem a frontendem. Pokud již máte existující kód napsaný v Pythonu pro backend, můžete ho znovu použít ve Skulptu pro frontend, čímž získáte jednotnou kódovou základnu a usnadníte údržbu a synchronizaci mezi oběma částmi aplikace.

Je třeba poznamenat, že Skulpt je interpretativním frameworkem, což znamená, že jeho výkon může být omezený ve srovnání s kompilovanými jazyky. To je důležité mít na paměti při vývoji webových aplikací s vysokými nároky na výkonost. Nejvhodnější je pro menší a středně velké webové projekty, které nevyžadují extrémně rychlou odezvu a výpočetní výkon.

Je důležité zmínit, že Skulpt je především určen pro menší projekty a prototypování. Pro větší a komplexnější webové aplikace s vyššími nároky na výkonost mohou být vhodnější jiné nástroje a frameworky. Skulpt se nevyhýbá omezením, která jsou spojena s interpretovanými jazyky, a může mít nižší výkon než jiné kompilované jazyky.

Celkově lze říci, že Skulpt je užitečným nástrojem pro vývoj webových aplikací pomocí Pythonu. Jeho schopnost překladu Python kódu na JavaScript a jeho interpretace v prohlížeči umožňuje vývojářům psát kód v Pythonu a spouštět ho přímo v webovém prostředí. Skulpt poskytuje podporu většiny funkcí a knihoven Pythonu, což umožňuje využívat znalosti a dovednosti v tomto jazyce při vývoji webových aplikací. [54] [55]

3.4 PyWebView

PyWebView je jednoduchá knihovna, která umožňuje vytvářet desktopové aplikace pomocí jazyka Python a webových technologií. Tato knihovna integruje komponentu prohlížeče (webview) do aplikace napsané v Pythonu, což umožňuje vytvářet uživatelská rozhraní pomocí HTML, CSS a JavaScriptu a současně využívat Python pro logiku aplikace. [56]

PyWebView poskytuje jednoduché a intuitivní rozhraní pro vytváření desktopových aplikací. Podporuje integraci s různými webovými enginy, jako je WebKit nebo EdgeHTML, a umožňuje zobrazovat webové stránky nebo vykreslovat vlastní uživatelská rozhraní pomocí webových technologií.

Tato knihovna je užitečná pro vývojáře, kteří mají zkušenosti s webovým vývojem a chtějí využít své znalosti HTML, CSS a JavaScriptu pro tvorbu desktopových aplikací. PyWebView poskytuje možnost využít množství dostupných knihoven a nástrojů pro webový vývoj při vytváření uživatelského rozhraní aplikace.

Díky integraci s Pythonem můžete v aplikaci využívat sílu a rozsáhlost jazyka Python pro manipulaci s daty, komunikaci se serverem, přístup k periferním zařízením atd. PyWebView je snadno rozšiřitelný a umožňuje vám psát a spouštět Python kód přímo v rámci webového rozhraní.

Celkově vzato, PyWebView je užitečný nástroj pro vývojáře, kteří preferují Python jako svůj hlavní programovací jazyk a chtějí vytvářet desktopové aplikace s moderním uživatelským rozhraním pomocí webových technologií. [57]

3.5 PyReact

PyReact je knihovna, která propojuje jazyk Python s populárním JS frameworkem React. Je navržena tak, aby umožnila vývojářům psát uživatelská rozhraní pomocí Python syntaxe, která se poté překládá na React komponenty. [58]

Funguje jako most mezi Pythonem a Reactem. Umožňuje vytvářet a manipulovat s React komponentami pomocí Python tříd a metod. Vývojáři mohou definovat uživatelská rozhraní ve formě Python kódu, který je přeložen na ekvivalentní React kód, který je spouštěn v prohlížeči.

S využitím PyReact mohou vývojáři využít sílu a jednoduchost Pythonu pro tvorbu uživatelského rozhraní v Reactu. Mohou využívat syntaxi Pythonu, podporované nástroje pro statickou analýzu kódu a další Python knihovny, které usnadňují vývoj.

Při použití PyReact musí být zajištěna kompatibilita mezi Python kódem a Reactem. PyReact se stará o překlad kódu na React komponenty a poskytuje rozhraní pro propojení Pythonu kódu s životním cyklem komponent v Reactu.

Jde o zajímavou volbu pro vývojáře, kteří mají zkušenosti s Pythonem a chtějí využít tuto znalost při vytváření uživatelských rozhraní s využitím Reactu. Je důležité mít na paměti, že PyReact je pouze rozhraním mezi Pythonem a Reactem a vyžaduje stále práci s Reactem a jeho ekosystémem.

3.6 RapydScript

RapydScript je jazyk a framework pro vývoj webových aplikací, který vychází z jazyka Python. Jeho hlavním cílem je poskytnout vývojářům možnost psát kód v Pythonu a převést ho na optimalizovaný JS, který je spustitelný ve webovém prostředí. RapydScript kombinuje sílu Pythonu s rychlostí a flexibilitou JS, což umožňuje tvorbu efektivních a výkonných webových aplikací.

Jednou z hlavních vlastností RapydScriptu je jeho syntaxe, která je velmi podobná Pythonu. To znamená, že vývojáři mohou využívat své znalosti Pythonu a psát kód ve způsobu, který je jim povědomý. Syntaxe je navržena tak, aby byla co nejpodobnější Pythonu, což zjednodušuje přechod a zvykání si na tento jazyk pro vývoj webových aplikací.

Poskytuje také širokou škálu funkcí a knihoven, které jsou běžné v Pythonu, jako jsou podmínky, smyčky, funkce, třídy a moduly. To umožňuje vývojářům psát kód ve způsobu, který je jim přirozený, a využívat mnoho známých konstrukcí a vzorců z Pythonu. Také nabízí podporu pro práci s DOM (Document Object Model), což umožňuje manipulaci s webovým obsahem, správu událostí a interakci s uživatelským rozhraním.

Další výhodou RapydScriptu je jeho rychlost a výkonost. Kód napsaný v RapydScriptu je přeložen na optimalizovaný JavaScript, který je spustitelný v prohlížeči. To znamená, že tyto webové aplikace mohou dosahovat vyššího výkonu a rychlosti ve srovnání s kódem napsaným v čistém JS. RapydScript se zaměřuje na generování efektivního a kompaktního kódu, což je výhodné zejména pro mobilní zařízení a pomáhá zlepšit uživatelskou zkušenost.

Je také velmi flexibilní a rozšiřitelný. Je možné integrovat další JS knihovny a frameworky do projektu a využívat jejich funkcionality společně s RapydScriptem. To umožňuje vývojářům využít širokou škálu dostupných nástrojů a rozšíření pro tvorbu webových aplikací. Podporuje také modulární strukturu kódu, což umožňuje rozdělení aplikace do samostatných modulů a opakované využití kódu.

Další výhodou je jeho snadná integrace s existujícími webovými technologiemi. Lze jej použít jako součást většího projektu, který již využívá HTML, CSS a JavaScript. Můžete importovat RapydScript soubory do svého projektu a využívat jeho funkcionality společně s ostatními technologiemi. Tím se umožní postupné zavádění RapydScriptu do existujícího kódu a sníží se náročnost na přechod na nový jazyk.

Je třeba si uvědomit, že se jedná o specializovaný nástroje pro vývoj webových aplikací a nenahrazuje plně Python. Existují některé odlišnosti a omezení v syntaxi a funkčnosti mezi Pythonem a RapydScriptem, a proto je třeba se seznámit s těmito rozdíly při vývoji aplikací. Celkově lze říci, že jde o užitečný framework a jazyk pro vývoj webových aplikací. Kombinuje sílu Pythonu s rychlostí a flexibilitou JS, což umožňuje vývojářům psát kód v Pythonu a převést ho na optimalizovaný JavaScript. RapydScript poskytuje podporu pro známé konstrukce Pythonu, umožňuje práci s DOM a nabízí rychlou a výkonnou generaci kódu. Je to užitečný nástroj pro vývojáře, kteří preferují syntaxi Pythonu a chtějí využívat jeho výhod při tvorbě webových aplikací. [59]

3.7 BeeWare

BeeWare je open-source projekt a framework zaměřený na vývoj multiplatformních aplikací v Pythonu. Jeho hlavním cílem je umožnit vývojářům psát kód v Pythonu a spouštět ho na různých platformách, včetně desktopových operačních systémů (Windows, macOS, Linux), mobilních zařízení (Android, iOS) a webových prohlížečů.

BeeWare poskytuje sadu nástrojů a knihoven, které umožňují vytvářet nativní aplikace s grafickým uživatelským rozhraním (GUI) pro různé platformy. Mezi tyto nástroje patří Toga, který poskytuje jednotné API pro vytváření uživatelského rozhraní napříč platformami. Toga je navržena tak, aby se co nejvíce podobala nativním prvkům GUI na jednotlivých platformách, což umožňuje vývojářům vytvářet aplikace, které vypadají a chovají se přirozeně na dané platformě.

BeeWare také zahrnuje nástroj Briefcase, který umožňuje balení aplikací do spustitelných souborů pro konkrétní platformu. Tím umožňuje snadnou distribuci a nasazení aplikací na různé platformy bez nutnosti instalace Pythonu nebo dalších závislostí.

Dalším důležitým prvkem BeeWare je BeeWare Toolbox, což je sada nástrojů pro správu a vývoj BeeWare projektů. Toolbox poskytuje přehlednou sadu nástrojů pro tvorbu a správu projektů, kompilaci kódu, spouštění testů, balení aplikací a další.

Je navržen tak, aby byl snadno rozšiřitelný a modularizovaný. Poskytuje možnost používat stávající knihovny Pythonu a rozšíření, což umožňuje vývojářům využívat bohatý ekosystém Pythonu při vývoji BeeWare aplikací.

Jednou z klíčových vlastností BeeWare je jeho zaměření na nativní výkon a uživatelskou zkušenost. Aplikace vytvořené pomocí BeeWare jsou optimalizovány pro konkrétní platformy a poskytují uživatelům pocit nativní aplikace.

Díky aktivní komunitě je BeeWare živým projektem a stále se vyvíjí. Komunita poskytuje dokumentaci, tutoriály a další zdroje, které pomáhají vývojářům při práci s BeeWare.

BeeWare je ideální volbou pro vývojáře, kteří chtějí psát kód v Pythonu a spouštět ho na různých platformách s důrazem na nativní uživatelskou zkušenost. Díky jednotnému API a nástrojům pro správu projektů a balení aplikací je vývoj multiplatformních aplikací snadný a efektivní.

Vzhledem k modulární architektuře BeeWare a jeho schopnosti využívat existující knihovny Pythonu je možné vytvářet robustní aplikace s bohatou funkcionalitou.

Celkově lze říci, že BeeWare je mocným frameworkem pro vývoj multiplatformních aplikací v Pythonu. Poskytuje jednoduché a konzistentní rozhraní pro vytváření uživatelského rozhraní, nástroje pro správu projektů a balení aplikací a podporu pro různé platformy. [60]

3.8 Kivy

Kivy je open-source framework napsaný v Pythonu, který je určen pro vývoj multiplatformních aplikací s grafickým uživatelským rozhraním (GUI). Jeho hlavním cílem je umožnit vývojářům psát kód v Pythonu a spouštět ho na různých platformách, včetně desktopových operačních systémů (Windows, macOS, Linux), mobilních zařízení (Android, iOS) a vestavěných systémů (Raspberry Pi)

Kivy poskytuje mnoho funkcí a nástrojů pro tvorbu aplikací s GUI. Obsahuje knihovny pro práci s grafikou, dotykovými událostmi, zvukem, animacemi a dalšími prvky GUI. Díky tomu lze vytvářet interaktivní uživatelská rozhraní s různými ovládacími prvky, jako jsou tlačítka, seznamy, textová pole a další.

Framework také podporuje událostmi řízené programování, což umožňuje reagovat na uživatelské akce, jako jsou kliknutí myši, stisknutí kláves, dotyky na obrazovce atd. Kivy nabízí také možnost animace prvků GUI a efekty přechodů mezi obrazovkami.

Kivy je navržen tak, aby byl snadno rozšiřitelný a modularizovaný. Nabízí množství rozšíření, které umožňují integrovat další funkce do aplikací, například práci se sítí, databázemi, senzory zařízení a další. K dispozici je také podpora pro použití jazyka kvůli rychlejšímu zobrazení nebo výpočtům.

Kromě toho Kivy poskytuje také sadu nástrojů pro tvorbu a správu projektů, testování aplikací, ladění a další vývojové operace. Podporuje různá vývojová prostředí, včetně populárních editorů Pythonu.

Díky svému multiplatformnímu charakteru a snadnému použití je Kivy populárním frameworkem pro vývoj aplikací s GUI v Pythonu. Umožňuje vývojářům vytvářet jednotné kódy pro různé platformy, což usnadňuje a urychluje vývoj aplikací. [61]

3.9 Streamlit

Streamlit je open-source framework pro rychlý vývoj interaktivních webových aplikací a vizualizací dat v Pythonu. Jeho hlavním cílem je umožnit vývojářům snadno vytvářet webové aplikace, které rychle reagují na změny v kódu a automaticky aktualizují uživatelské rozhraní.

Zaměřuje se na jednoduchost použití a minimalismus. Je navržen tak, aby vývojářům umožnil rychle vytvořit funkční webovou aplikaci pouze pomocí několika řádků kódu. Vývojáři mohou jednoduše importovat a použít různé knihovny Pythonu pro analýzu dat, strojového učení, vizualizace atd., a Streamlit se postará o automatické vykreslení uživatelského rozhraní na webovou stránku.

Poskytuje jednoduché API pro různé typy interaktivních ovládacích prvků, jako jsou tlačítka, vstupní pole, výběrové seznamy atd., což umožňuje vývojářům snadno vytvářet interaktivní prvky ve svých aplikacích. Uživatelé mohou interagovat s aplikací prostřednictvím webového rozhraní a vidět okamžité změny v reakci na své akce

Také nabízí možnost jednoduché vizualizace dat. Vývojáři mohou pomocí různých knihoven pro vizualizaci, jako je Matplotlib nebo Plotly, vytvářet grafy, diagramy, interaktivní vizualizace a další. Tyto vizualizace lze snadno integrovat do Streamlit aplikace a zobrazit je uživatelům.

Další výhodou Streamlit je jeho schopnost automaticky sledovat změny v kódu a okamžitě aktualizovat webovou aplikaci. To umožňuje vývojářům rychle experimentovat s různými

konfiguracemi, parametry a daty a vidět okamžité výsledky bez nutnosti manuálního opakovaného spouštění aplikace.

Je navržen pro snadnou nasazení na různé platformy, včetně lokálního prostředí, serverů v cloudu nebo sdílení aplikace prostřednictvím webových služeb. Je velmi populární mezi vývojáři datového vědeckého a strojového učení, protože umožňuje rychlé prototypování a sdílení výsledků svých analýz a modelů.

Celkově lze říci, že Streamlit je mocným nástrojem pro rychlý vývoj interaktivních webových aplikací v Pythonu. Jeho jednoduché rozhraní, podpora vizualizace dat a automatická aktualizace v reakci na změny v kódu jej činí oblíbeným mezi datovými vědci, vývojáři strojového učení a dalšími, kteří chtějí rychle sdílet své analýzy a aplikace s ostatními. [62]

3.10 Pyjs

Pyjs je open-source framework, který umožňuje vývojářům psát webové aplikace v jazyce Python, které jsou následně přeloženy do jazyka JavaScript a spouštěny ve webovém prohlížeči. Tím umožňuje vývojářům využívat výhod a pohodlí programovacího jazyka Python pro vývoj webových aplikací.

Hlavní myšlenkou Pyjs je překlad Python kódu do JavaScriptu, který je plně funkční ve všech moderních webových prohlížečích. Vývojáři mohou tedy psát kód v Pythonu a využívat jeho syntaxe, datových typů, funkcí a modulů, které jsou přeloženy do ekvivalentního kódu v JavaScriptu. To umožňuje využívat silné stránky Pythonu, jako je čistý a čitelný kód, snadné spravování závislostí a široká komunita, při vývoji webových aplikací.

nabízí také sadu knihoven a nástrojů, které usnadňují vývoj webových aplikací. Mezi tyto nástroje patří například Pyjamas Desktop, který umožňuje vytvářet desktopové aplikace pomocí Pyjs, nebo Pyjsdl, což je implementace knihovny Pygame pro Pyjs, která umožňuje vývoj her a multimediálních aplikací. Díky těmto nástrojům je možné rozšiřitelnost a flexibilita Pyjs ještě více rozšířena.

Jednou z hlavních výhod Pyjs je možnost sdílet kód mezi klientskou a serverovou částí aplikace. Vývojáři mohou psát společný kód v Pythonu, který je poté přeložen do JS pro běh ve webovém prohlížeči a také může být spuštěn na serveru pomocí Python interpreteru. To

umožňuje vývojářům maximalizovat znovupoužitelnost kódu a snížit nároky na údržbu aplikace.

Překlad Pythonu do JS může mít určité omezení a výkonnostní kompromisy. Mají různé vlastnosti a chování, a proto se může stát, že některé funkce Pythonu nebudou plně podporovány v Pyjs nebo může být vyžadována dodatečná optimalizace kódu pro dosažení požadovaného výkonu. Je důležité provést testování a ověřit, zda přeložený kód v Pyjs správně funguje a odpovídá očekáváním.

Výhody Pyjs spočívají v tom, že umožňuje vývojářům využívat sílu a pohodlí jazyka Python při tvorbě webových aplikací. Díky překladu Pythonu do JS je možné psát kód v Pythonu a spouštět ho ve webovém prohlížeči. Tím je zajištěna efektivní a jednoduchá cesta k vývoji webových aplikací s použitím oblíbeného jazyka Python.

Celkově lze tedy říci, že Pyjs je zajímavým frameworkem pro vývoj webových aplikací v jazyce Python. Poskytuje vývojářům možnost psát kód v Pythonu a přeložit ho do JS pro spouštění ve webovém prohlížeči. Při výběru Pyjs je však důležité zvážit omezení a rozdíly mezi Pythonem a JS, sledovat aktuálnost a podporu frameworku a důkladně testovat přeložený kód, aby byla zajištěna správná funkčnost a výkon aplikace. [63]

3.11 Jupyter Notebook

Jupyter Notebook je open-source webové prostředí pro interaktivní výpočet a vizualizaci dat. Jeho hlavním cílem je umožnit vědcům, vývojářům a datovým analytikům vytvářet a sdílet dokumenty obsahující kód, výsledky výpočtů, vizualizace a popisy.

Jupyter Notebook je založen na konceptu buněk, které umožňují psát a spouštět kód v jednotlivých blocích. Existují dva hlavní typy buněk: kódové buňky a textové buňky. Kódové buňky slouží k psaní a spouštění kódu v různých programovacích jazycích, včetně Pythonu, R, Julia, Scala a dalších. Textové buňky obsahují popisy, formátovaný text nebo matematické rovnice napsané v systému LaTeX.

Jupyter Notebook poskytuje prostředí, ve kterém je možné psát kód, spouštět ho a okamžitě vidět výsledky. Výpočty mohou být prováděny postupně v jednotlivých buňkách, což umožňuje systematický a iterativní přístup k vývoji a analýze dat. Díky tomu je možné snadno zkoumat datové sady, provádět statistické analýzy, vizualizovat data a dokumentovat celý proces.

Jupyter Notebook umožňuje také bohatou vizualizaci dat. Výsledky výpočtů mohou být prezentovány pomocí grafů, tabulek, interaktivních ovládacích prvků a dalších vizuálních prvků. Tím se zlepšuje schopnost komunikace a prezentace datových analýz a výsledků výzkumu.

Další výhodou Jupyter Notebook je možnost sdílet vytvořené dokumenty s ostatními. Jupyter Notebook soubory mohou být snadno exportovány do různých formátů, včetně HTML, PDF nebo Markdown. To umožňuje sdílet výsledky, výzkumné zprávy a tutoriály s kolegy, klienty nebo širší komunitou.

Jupyter Notebook je také integrován s mnoha dalšími nástroji a knihovnami, které rozšiřují jeho funkce. Existuje rozsáhlá ekosystém rozšíření, které umožňují například integraci s verzovacími systémy, práci s Big Data, tvorbu interaktivních dashboardů a další.

Celkově lze říci, že Jupyter Notebook je velmi užitečný nástroj pro interaktivní vývoj, analýzu dat a sdílení výsledků. Jeho jednoduché rozhraní, podpora různých programovacích jazyků a možnost bohaté vizualizace dat ho činí oblíbenou volbou mezi vědci, vývojáři a analytiky. [64]

II. PRAKTICKÁ ČÁST

4 POUŽITÉ TECHNOLOGIE

Pro vývoj ukázkové aplikace byl použit framework Django, který poskytuje podporu pro práci s databází SQLite. Django je založen na interpretovaném jazyce Python, který je mezi vývojářskou komunitou velmi oblíbený díky své přívětivosti a přehlednosti.

Aplikace se skládá z frontendové a backendové části. Backendová část, která je implementována v rámci frameworku Django, zajišťuje správu databáze, logiku aplikace a poskytuje API pro komunikaci s frontendovou částí. Díky podpoře ORM (Object-Relational Mapping) je práce s databází jednoduchá a efektivní. Standardně je používána databáze SQLite, která je snadno nasaditelná a není potřeba konfigurace.

Frontendová část aplikace sestává z HTML, CSS a JavaScript souborů. Tyto soubory slouží k vizualizaci a interakci s uživatelem. Kromě toho se v aplikaci využívá také PyScript, což je knihovna umožňující kreslení grafiky pomocí HTML5. PyScript poskytuje funkce pro vytváření a manipulaci s různými tvary, nastavování barev, tloušťky čar a další.

V rámci ukázkové aplikace je využita síla a možnosti Pythonu ve spojení s frameworkem Django pro implementaci backendové logiky a práci s databází. PyScript pak přidává interaktivní a grafickou část aplikace, umožňující uživatelům kreslit různé tvary a experimentovat s grafikou přímo v prohlížeči.

Tato kombinace Django, Pythonu a PyScriptu poskytuje uživatelům přívětivé a výkonné prostředí pro vývoj webových aplikací s možností práce s databází a interaktivním vykreslováním grafiky.

4.1 Django

Django je open-source webový framework napsaný v jazyce Python. Vznikl v roce 2005 a od té doby se stal jedním z nejpopulárnějších frameworků pro vývoj webových aplikací. Django se vyznačuje svou robustností, flexibilitou a rozšiřitelností, a je často využíván pro vývoj komplexních a škálovatelných webových projektů.

Jednou z hlavních výhod Django je jeho architektura založená na modelu MVC (Model-View-Controller) nazvaná MTV (Model-View-Template). Tato architektura odděluje logiku aplikace, datový model a uživatelské rozhraní, což usnadňuje správu a údržbu kódu. Django také poskytuje bohatou sadu funkcí a nástrojů, které urychlují vývoj a snižují potřebu opakovaného psaní kódu.

Jedním z nejvýznamnějších prvků Django je jeho ORM (Object-Relational Mapping) vrstva. ORM umožňuje vývojářům pracovat s databází pomocí objektově orientovaného přístupu, což zjednodušuje manipulaci s daty a snižuje potřebu psát SQL dotazy manuálně. ORM podporuje různé typy databází, včetně MySQL, PostgreSQL, SQLite a dalších.

Django také poskytuje vestavěnou správu uživatelského rozhraní, která zahrnuje autentizaci, autorizaci, správu uživatelů a další. To usnadňuje implementaci funkcí jako registrace uživatelů, přihlášení, správa profilu a práva přístupu. Django také podporuje mezinárodní lokalizaci a umožňuje jednoduchou tvorbu vícejazyčných aplikací.

Další výhodou Django je jeho silná podpora pro tvorbu RESTful API. Django poskytuje nástroje a knihovny pro vytváření API, které umožňují komunikaci s webovými službami a mobilními aplikacemi. Tím je Django vhodný pro vývoj moderních webových aplikací, které vyžadují rozhraní pro interakci s jinými systémy.

Další zajímavou funkcí je tzv. automatická administrace, která umožňuje snadnou tvorbu administrátorského rozhraní pro správu dat v aplikaci. Administrátorské rozhraní lze přizpůsobit dle potřeb a lze v něm snadno vytvářet, upravovat a mazat data.

Další výhodou je velké množství knihoven a rozšiřujících balíčků, které vývojáři mohou použít pro zjednodušení a urychlení vývoje. Má také celkově velmi dobrou dokumentaci a aktivní komunitu, což usnadňuje získání pomoci a nápadů při řešení problémů.

Mezi nevýhody Django patří poměrně složitá konfigurace a inicializace projektu, což může být pro začátečníky zpočátku obtížné. Vyžaduje poměrně robustní infrastrukturu, což může být problém pro menší projekty nebo pro ty, kteří nechtějí mít složitou infrastrukturu. Vývojáři také musí být obeznámeni s architekturou a koncepty Django, aby mohli efektivně pracovat s touto platformou.

Celkově lze říci, že Django je vynikající volba pro vývoj webových aplikací v Pythonu. Má vysokou úroveň flexibility, škálovatelnosti a bezpečnosti. Jeho administrátorské rozhraní, velké množství rozšiřujících balíčků a aktivní komunita z něj činí jednu z nejlepších platform pro vývoj webových aplikací. [65]

4.2 HTML5

HTML5 je pátou verzí značkovacího jazyka HTML (Hypertext Markup Language), který se používá k tvorbě webových stránek. Byl vyvinut s cílem poskytnout bohatší možnosti pro

tvorbu webového obsahu a zlepšit interakci s uživateli. HTML5 přináší mnoho nových prvků a funkcí, které umožňují vytvářet moderní a dynamické webové stránky.

Jednou z klíčových vlastností HTML5 je podpora multimédií bez nutnosti používání zásuvných modulů. Tyto moduly mohou obsahovat bezpečnostní trhliny a ohrožovat uživatele. Nové prvky jako `<video>` a `<audio>` umožňují snadno vkládat a přehrávat video a zvukové soubory přímo ve webovém prohlížeči. Díky tomu se zlepšuje uživatelská interakce a zvyšuje se přístupnost webového obsahu.

Další významnou částí je `canvas` (`<canvas>`), což je prázdné plátno, které umožňuje programovatelné vykreslování obrázků, grafiky a animací pomocí JavaScriptu. Canvas poskytuje vývojářům mnohem větší volnost při tvorbě a manipulaci s grafikou na webových stránkách a umožňuje vytvářet bohaté vizuální efekty.

HTML5 také přináší nové formulářové prvky, které usnadňují a zlepšují interakci uživatele se stránkou. Patří sem například `<input type="date">` pro výběr data, `<input type="email">` pro zadávání e-mailové adresy nebo `<input type="range">` pro posuvníky.

Rozšířením HTML5 jsou také tzv. geolokační API, která umožňují webovým stránkám získávat informace o poloze uživatele. Tato funkce je často využívána v mobilních aplikacích nebo při vyhledávání blízkých obchodů či restaurací.

HTML5 dále poskytuje podporu pro offline aplikace, které umožňují ukládat a přistupovat k datům i bez připojení k internetu. Tímto způsobem lze vytvářet webové aplikace, které fungují podobně jako klasické desktopové programy.

Kromě všech výše uvedených novinek HTML5 také zavádí mnoho dalších vylepšení, jako je například podpora nových typů sémantických elementů (`<header>`, `<footer>`, `<main>`, `<nav>`, `<article>`, atd.), které usnadňují strukturování obsahu webových stránek a zlepšují přístupnost pro uživatele a vyhledávače.

HTML5 přináší mnoho inovací a vylepšení, které umožňují vytvářet moderní a interaktivní webové stránky. Díky implementovaným prvkům, funkcím a API je možné vytvářet bohatý multimediální obsah, grafiku, animace a využívat pokročilé možnosti interakce s uživatelem. HTML5 přispívá k lepšímu uživatelskému zážitku, vyšší přístupnosti a větší flexibilitě při tvorbě webového obsahu. S jeho využitím je možné vytvořit moderní webové stránky a aplikace, které odpovídají současným standardům a potřebám uživatelů. [66]

4.3 Pip

Pip neboli Package Installer for Python je standardní nástroj pro správu balíčků v jazyce Python. Jeho hlavním úkolem je umožnit snadnou instalaci, aktualizaci a odinstalaci balíčků a jejich závislostí, což usnadňuje práci vývojářů a správu projektů. Bez tohoto nástroje se většina vývojářů neobejde.

Pip je součástí oficiální distribuce Pythonu od verze 3.4. Původně byl inspirován nástrojem `easy_install` a postupem času se stal de-facto standardem pro správu balíčků v ekosystému Pythonu. Umožňuje uživatelům jednoduše vyhledávat, stahovat a instalovat balíčky z veřejných repozitářů, jako je Python Package Index (PyPI), který je největším repozitářem balíčků pro Python.

Pip také podporuje instalaci balíčků z různých zdrojů, včetně lokálních souborů, Git repozitářů nebo URL adres. To poskytuje flexibilitu vývojářům, kteří mohou pracovat s vlastními balíčky nebo testovat nejnovější verze zdrojových kódů.

Další výhodou Pipu je jeho schopnost automaticky řešit závislosti balíčků. To znamená, že při instalaci určitého balíčku se také automaticky nainstalují všechny potřebné závislosti. Tím se zjednodušuje proces správy závislostí a minimalizuje riziko konfliktů mezi různými verzemi balíčků.

Jedním z hlavních problémů je, že pip nedokáže automaticky spravovat konflikty mezi balíčky, pokud jsou závislé na různých verzích stejného balíčku. To může vést k problémům s funkčností aplikace a vyžadovat manuální řešení konfliktů.

Dalším aspektem je, že pip se spoléhá na veřejné repozitáře balíčků, jako je PyPI. To znamená, že dostupnost a spolehlivost balíčků závisí na jejich udržování a aktualizaci ze strany komunity. Někdy mohou být balíčky nedostupné, nekompatibilní nebo zastaralé, což může zpomalit vývojový proces.

Nicméně, přes tyto omezení je Pip stále klíčovým nástrojem pro správu balíčků v Pythonu a poskytuje vývojářům efektivní způsob, jak spravovat závislosti a rychle nasazovat potřebné balíčky do svých projektů. [67]

5 PRAKTICKÉ UKÁZKY TUTORIÁLU

Byl vytvořen podrobný tutoriál, který demonstruje základní funkcionality frameworku PyScript. Tutoriál obsahuje celkem sedm ukázek, které postupují od jednoduchých úloh nastavení až po složitější skripty, které plně využívají potenciál PyScriptu. Každý kód je prezentován v čitelné formě a v textu jasně odlišen. Je doprovázen vysvětlením, které usnadňuje pochopení a umožňuje i začínajícím vývojářům získat hluboké znalosti pro ovládnutí tohoto frameworku.

Tutoriál se zaměřuje na jednoduché a přehledné vysvětlení principů a technik PyScriptu. Každá stránka tutoriálu je strukturována s navigační lištou, nadpisem, krátkými kódy a vysvětlením, které popisuje, co daný kód přesně dělá. Tímto způsobem se snaží usnadnit uživatelům orientaci a umožnit jim postupně se seznámit s různými aspekty PyScriptu.

Díky tomuto tutoriálu mají vývojáři přístup k praktickým příkladům a jsou schopni se detailně seznámit s různými funkcemi a možnostmi, které PyScript nabízí. Tutoriál se snaží být nápomocný a poskytuje základní vzdělání a přehled o tom, jak efektivně pracovat s PyScriptem.

Stručný přehled, co nalezneme na jednotlivých stranách:

1. Úvodní strana – charakteristika PyScriptu
2. První strana – Základní nastavení
3. Druhá strana – Pokročilejší nastavení
4. Třetí strana – Práce s pyrepl a pyterminal
5. Čtvrtá strana – Práce s funkcí display
6. Pátá strana – Manipulace s DOMem přes Element API
7. Šestá strana – Načítání souboru z disku
8. Sedmá strana – Využití kamery v browseru
9. Editor – Možnost psaní Python kódu a jeho exekuce

5.1 Úvodní strana

Úvodní stránka mapuje, co vlastně PyScript je a čeho se snaží dosáhnout. Obsahuje jeho stručnou charakteristiku. Uvádí základní nutné prerekvizity a znalosti potřebné

k bezproblémové a efektivní práci. Jde především o základní znalosti, které vývojář potřebuje mít.

Úvodní strana Začni tutorial Vyzkoušej PyScript kód

Framework PyScript

Co je to PyScript?

PyScript je framework, který umožňuje uživatelům vytvářet komplexní aplikace v jazyce Python v prohlížeči s využitím rozhraní HTML. PyScript stojí na Pyodide, WASM a moderních webových technologiích. Framework PyScript poskytuje uživatelům na všech úrovních zkušeností přístup k expresivnímu, snadno naučitelnému programovacímu jazyku s nespočtenými možnostmi využití.

K dispozici je jednoduchý tutorial, jak začít programovat v PyScriptu a plně využít jeho možnosti.

Co PyScript přináší?

- Síla jazyka Python přímo v prohlížeči bez nutnosti mít backend
- Přímá dostupnost různých balíčků pro práci s daty, grafy, obrázky a další
- Podpora dvousměrné komunikace mezi JavaScriptem a PyScriptem umožňující předávání informací mezi objekty
- Plně konfigurovatelné prostředí pro specifické potřeby aplikace
- Připravené designové prvky pro rychlejší vývoj
- Možnost vytváření vlastních balíčků podporující požadovanou funkcionalitu a snadnou použitelnost spolu s PyScriptem

Prerekvizity

- Znalost jazyků Python a JavaScript
- Základní znalost HTML a CSS
- Prohlížeč, nejlépe Chrome, dle doporučení tvůrců PyScriptu

Odkaz na oficiální stránky:

[PyScript](#)

Obrázek 10 – Úvodní strana tutorialu

Navigační panel slouží k přepínání jednotlivých stránek. Na obrázku je patrné, že odkazuje na začátek tutorialu a na editor spustitelného kódu v prohlížeči.

5.2 Ukázka 1

Na první stránce se nachází základní nutné nastavení pro práci s dotyčným frameworkem. Na začátku je uveden základní kámen aplikace, a to je html soubor. Tento soubor je postupně rozšiřován až do plnohodnotné podoby funkčního skriptu demonstrující funkcionalitu. V několika ukázkách je postupně popsáno, jak je potřeba nastavit jednotlivé komponenty zajišťující hladký běh aplikace.

[Úvodní strana](#) [Pokračuj v tutoriálu](#) [Vyzkoušej PyScript kód](#)

Tutoriál strana první - Základní nastavení

Nejprve si vytvoříme prázdný HTML5 soubor, se kterým budeme pracovat.

```
<!DOCTYPE html>
  <html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Title</title>
  </head>
  <body>

  </body>
</html>
```

Následně vložíme odkazy na css soubory a samotný zdrojový soubor PyScriptu.

```
<!DOCTYPE html>
  <html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>PyScript tutoriál</title>
    <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
    <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  </head>
```

Obrázek 11 – První strana tutoriálu

5.3 Ukázka 2

Na druhé straně rozšiřujeme stávající soubor o nové tagy složitějšího nastavení. Toto nové nastavení umožňuje rozšíření o novou funkcionalitu. Ta sestává z upravení konfigurace frameworku. Nastavuje se načítací stránka, zvolený jazykový interpreter a případně se načítají doplňkové soubory.

[Úvodní strana](#) [Zpět v tutoriálu](#) [Pokračuj v tutoriálu](#) [Vyzkoušej PyScript kód](#)

Tutoriál strana druhá - Pokročilejší nastavení

Pokračujeme s již vytvořeným html souborem a přidáme nové tagy <py-config>.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>PyScript tutoriál</title>
  <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
  <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  <py-config>

  </py-config>
</head>
<body>
  <pyscript>
    print("První text z jazyka Python v prohlížeči!")
  </pyscript>
</body>
</html>
```

Vložíme nové konfigurační nastavení, které způsobí, že se po načtení prostředí automaticky zavře načítací okno a nastaví si náš požadovaný interpreter, který chceme používat. Tento zápis odpovídá formátu toml, který je brán jako hlavní, pokud není specifikováno jinak.

```
<!DOCTYPE html>
<html lang="en">
```

Obrázek 12 – Druhá strana tutoriálu

5.4 Ukázka 3

Stránka třetí se věnuje fungování smyčky vyhodnocování kódu a terminálu zobrazující se přímo v browseru. Objasňuje, jak se obě komponenty chovají a jak je potřeba k nim přistupovat. Při dodržení daných doporučení je správná funkce téměř zaručená.

[Úvodní strana](#) [Zpět v tutoriálu](#) [Pokračuj v tutoriálu](#) [Vyzkoušej PyScript kód](#)

Tutoriál strana třetí - Práce s py-repl a py-terminal

Pokračujeme s již vytvořeným html souborem a přidáme nový tag `<py-repl>`, který vytvoří interaktivní smyčku pro vyhodnocení vloženého kódu.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>PyScript tutoriál</title>
  <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
  <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  <py-config type="json">
    {
      "packages": ["numpy", "matplotlib"],
      "splashscreen": {
        "autoclose": true
      },
      "interpreters": [{
        "src": "https://cdn.jsdelivr.net/pyodide/v0.21.2/full/pyodide.js",
        "name": "pyodide-0.21.2",
        "lang": "python"
      }],
    }
  </py-config>
</head>
<body>
  <pyscript>
    print("První text z jazyka Python v prohlížeči!")
  </pyscript>
  <py-repl auto-generate="true"></py-repl>
</body>
```

Obrázek 13 – Třetí strana tutoriálu

5.5 Ukázka 4

Stránka čtvrtá se zaměřuje na práci s funkcí `display`. Tato funkce umožňuje zobrazení textu, grafů a obrázků. Kód pracuje a vykresluje čitelný a přehledný graf.

[Úvodní strana](#) [Zpět v tutoriálu](#) [Pokračuj v tutoriálu](#) [Vyzkoušej PyScript kód](#)

Tutoriál strana čtvrtá - Práce s funkcí display

Pokračujeme s již vytvořeným html souborem. Přidáme do `<py-script>` tagu nový kód. Můžeme vymazat tagy `<py-repl>` a `<py-terminal>`. Přidaný kód vykreslí pomocí balíčku `matplotlib` a `numpy` graf.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>PyScript tutoriál</title>
  <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
  <script defer src="https://pyscript.net/latest/pyscript.js"></script>
<py-config type="json">
  {
    "packages": ["numpy", "matplotlib"],
    "splashscreen": {
      "autoclose": true
    },
    "interpreters": [{
      "src": "https://cdn.jsdelivr.net/pyodide/v0.21.2/full/pyodide.js",
      "name": "pyodide-0.21.2",
      "lang": "python"
    }],
  }
</py-config>
</head>
<body>
  <pyscript>
    import matplotlib.pyplot as plt
    import numpy as np
```

Obrázek 14 – Čtvrtá strana tutoriálu

5.6 Ukázka 5

Na straně páté je uvedena základní manipulace s jednotlivými elementy DOMu pomocí Element API.

[Úvodní strana](#) [Zpět v tutoriálu](#) [Pokračuj v tutoriálu](#) [Vyzkoušej PyScript kód](#)

Tutoriál strana pátá - Manipulace DOMu za pomoci Element API

Pokračujeme s již vytvořeným html souborem. Kam vložíme následující kód, který vezme hodnotu z inputu, kde jsme definovali jeho hodnotu a tu vypíše do terminálu.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>PyScript tutoriál</title>
  <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
  <script defer src="https://pyscript.net/latest/pyscript.js"></script>
<py-config type="json">
  {
    "packages": ["numpy", "matplotlib"],
    "splashscreen": {
      "autoclose": true
    },
    "interpreters": [{
      "src": "https://cdn.jsdelivr.net/pyodide/v0.21.2/full/pyodide.js",
      "name": "pyodide-0.21.2",
      "lang": "python"
    }],
  }
</py-config>
</head>
<body>
  <input id="custom-input" value="Value from input"></input>
  <pyscript>
    custom_input = Element('custom-input')
    print(custom_input.value)
  </pyscript>
```

Obrázek 15 – Pátá strana tutoriálu

5.7 Ukázka 6

Ukázka na stránce šesté už aplikuje složitější a zajímavější funkcionalitu. Jedná se o načtení obrázku z disku přímo do prostředí prohlížeče, se kterým můžeme eventuálně dále pracovat.

Úvodní strana Zpět v tutoriálu Pokračuj v tutoriálu Vyzkoušej PyScript kód

Tutoriál strana šestá - Načítání obrázku z disku a jeho zobrazení

Vložíme tento kód a přes input přidáme obrázek, který se následně zobrazí na stránce.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>PyScript tutoriál</title>
  <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
  <script defer src="https://pyscript.net/latest/pyscript.js"></script>
<py-config type="json">
  {
    "packages": ["numpy", "matplotlib"],
    "splashscreen": {
      "autoclose": true
    },
    "interpreters": [{
      "src": "https://cdn.jsdelivr.net/pyodide/v0.21.2/full/pyodide.js",
      "name": "pyodide-0.21.2",
      "lang": "python"
    }],
  }
</py-config>
</head>
<body>
  <label for="Upload an image"></label><input type="file" id="file-upload">
  <pyscript>
  <div id="output_upload"></div>
  from js import document, console, window
```

Obrázek 16 – Šestá strana tutoriálu

5.8 Ukázka 7

Na poslední straně tutoriálu se nachází kód umožňující focení přes webkameru a následně se tato fotka zobrazí v prohlížeči.

[Úvodní strana](#) [Zpět v tutoriálu](#) [Na začátek tutoriálu](#) [Vyzkoušej PyScript kód](#)

Tutoriál strana sedmá - Využití kamery v browseru

Vložíme tento kód a následně můžeme pořídit foto přes webkameru a vložit do browseru.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>PyScript tutoriál</title>
  <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
  <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  <py-config type="json">
    {
      "packages": ["numpy", "matplotlib"],
      "splashscreen": {
        "autoclose": true
      },
      "interpreters": [{
        "src": "https://cdn.jsdelivr.net/pyodide/v0.21.2/full/pyodide.js",
        "name": "pyodide-0.21.2",
        "lang": "python"
      }],
    }
  </py-config>
</head>
<body>

  <button id="start-camera">Start Camera</button>
  <button id="click-photo">Take Photo</button><br>
  <video id="video" width="600" height="400" autoplay></video>
```

Obrázek 17 – Sedmá strana tutoriálu

5.9 Editor

Jako poslední je uveden interaktivní editor, kde si může uživatel vyzkoušet kód z předchozích ukázek a otestovat jeho funkčnost a správnost. Oblast pro psaní kódu je nastavená na automatické přidání nové instance po úspěšné exekuci předchozího napsaného kódu.

PyScript Playground

Zde můžete vyzkoušet základní funkcionalitu.

Stačí napsat libovolný kód v Pythonu a PyScript se o jeho exekuci postará.

Kód lze spustit kliknutím na zelenou šipku, která se objeví v pravém dolním rohu po najetí kurzorem myši do této oblasti.

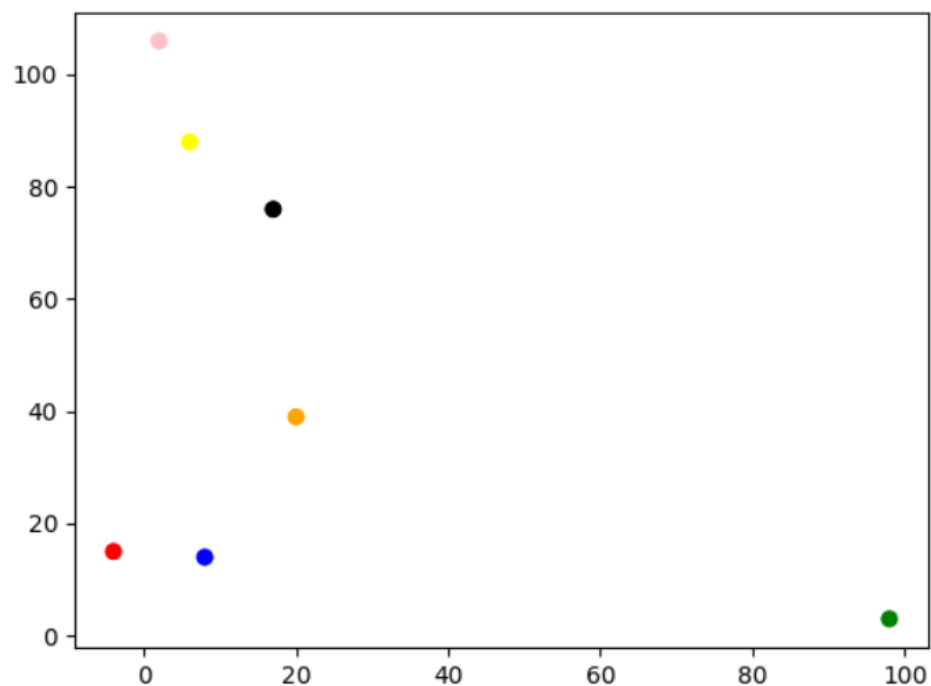
```
1 print("Nejlepší tutoriál :-)")
```

```
1
```

```
Nejlepší tutoriál :-)
```

Obrázek 18 – První ukázka z editoru

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 x = np.array([-4,98,8,6,2,17,20])
5 y = np.array([15,3,14,88,106,76,39])
6 colors = np.array(["red","green","blue","yellow","pink","black","orange"])
7 plt.scatter(x, y, c=colors)
8 plt
```



Obrázek 19 – Druhá ukázka z editoru

6 ZHODNOCENÍ FRAMEWORKU PYSCRIPT

PyScript přináší jednoznačnou výhodu, kterou je možnost přesunout logiku aplikace z backendu na frontend, což může výrazně usnadnit vývoj a ušetřit čas. Nicméně, tato výhoda je spojena s určitými omezeními a kompromisy.

Python jako programovací jazyk je známý svou náročností na paměť a knihovny psané v Pythonu mohou být velké a obsáhlé. To znamená, že při použití PyScriptu je nutné stáhnout a načíst do prohlížeče větší objem dat než při použití jiných frameworků. Velikost stahovaných balíčků závisí na počtu a rozsahu knihoven potřebných pro vývoj konkrétní aplikace.

Celkově ale výhody značně převažují nad nevýhodami a PyScript má do budoucna velký potenciál stát se velmi rozšířeným a oblíbeným frameworkem.

6.1 Stahované balíčky a nástroje

Níže uvedené obrázky z vývojářského prostředí v prohlížeči v sekci Network ilustrují proces stahování balíčků a dalších potřebných nástrojů pro správnou funkci PyScriptu. Tyto obrázky poskytují přehled o tom, jaká data jsou přenášena mezi serverem a prohlížečem během inicializace a spuštění PyScriptu.

Je důležité mít na paměti, že velikost a objem dat ke stažení se může lišit v závislosti na konkrétním projektu a použitých knihovnách. Přestože toto může být překážkou zejména při inicializaci aplikace, pokud jsou balíčky již staženy, další interakce s aplikací mohou být plynulé a efektivní.

Resource	Size	Type	Method	Time
editor/	200	document	Other	1.5 kB 6 ms
pyscript.css	200	stylesheet	(index)	2.0 kB 14 ms
pyscript.js	200	script	(index)	287 kB 74 ms
pyodide.js	200	script	main.ts:138	7.1 kB 23 ms
pyodide_py.tar	200	fetch	compat.ts:153	165 kB 39 ms
pyodide.asm.js	200	script	compat.ts:181	253 kB 43 ms
pyodide.asm.data	200	wasm	pyodide.asm.js:10	3.4 MB 177 ms
repdata.json	200	fetch	pyodide.asm.js:10	14.1 kB 18 ms
pyodide.asm.wasm	200	wasm	pyodide.asm.js:10	2.8 MB 180 ms
micropip-0.2.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	36.4 kB 8 ms
pparsing-3.0.9-py3-none-any.whl	200	fetch	pyodide.asm.js:10	98.6 kB 10 ms
packaging-21.3-py3-none-any.whl	200	fetch	pyodide.asm.js:10	40.9 kB 9 ms
json	200	fetch	pyodide.asm.js:10	36.8 kB 16 ms
requests-2.30.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	63.2 kB 15 ms
json	200	fetch	pyodide.asm.js:10	12.8 kB 15 ms
json	200	fetch	pyodide.asm.js:10	103 kB 17 ms
json	200	fetch	pyodide.asm.js:10	25.6 kB 15 ms
idna-3.4-py3-none-any.whl	200	fetch	pyodide.asm.js:10	61.9 kB 16 ms
charset_normalizer-3.1.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	46.8 kB 15 ms
urllib3-2.0.2-py3-none-any.whl	200	fetch	pyodide.asm.js:10	124 kB 17 ms
matplotlib-3.5.2-cp310-cp310-emscrip..._3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	6.5 MB 593 ms
cycler-0.11.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	6.5 kB 16 ms
six-1.16.0-py2.py3-none-any.whl	200	fetch	pyodide.asm.js:10	11.2 kB 19 ms
distutils-1.0.0.zip	200	fetch	pyodide.asm.js:10	269 kB 51 ms

Obrázek 20 – Obrázek z vývojářského prostředí ze sekce Network I

Status	Package Name	Size	Type	Source	Size	Time	Progress
☐	fonttools-4.38.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	967 kB	128 ms	■
☐	kwisolver-1.4.4-cp310-cp310-emscripten_3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	34.8 kB	44 ms	■
☐	numpy-1.23.5-cp310-cp310-emscripten_3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	3.1 MB	384 ms	■
☐	PIL-9.1.1-cp310-cp310-emscripten_3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	948 kB	219 ms	■
☐	python-dateutil-2.8.2-py2.py3-none-any.whl	200	fetch	pyodide.asm.js:10	248 kB	170 ms	■
☐	pytz-2022.7-py2.py3-none-any.whl	200	fetch	pyodide.asm.js:10	500 kB	248 ms	■
☐	matplotlib_pyodide-0.1.1-py3-none-any.whl	200	fetch	pyodide.asm.js:10	18.7 kB	168 ms	■
☐	pandas-1.5.2-cp310-cp310-emscripten_3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	5.3 MB	1.21 s	■
☐	setuptools-65.6.3-py3-none-any.whl	200	fetch	pyodide.asm.js:10	882 kB	465 ms	■
☐	bokeh-3.0.3-py3-none-any.whl	200	fetch	pyodide.asm.js:10	16.5 MB	2.19 s	■
☐	Jinja2-3.1.2-py3-none-any.whl	200	fetch	pyodide.asm.js:10	133 kB	558 ms	■
☐	MarkupSafe-2.1.1-cp310-cp310-emscripten_3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	12.2 kB	355 ms	■
☐	typing_extensions-4.4.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	26.8 kB	377 ms	■
☐	PyYAML-6.0-cp310-cp310-emscripten_3_1_27_wasm32.whl	200	fetch	pyodide.asm.js:10	116 kB	623 ms	■
☐	pyodide_http-0.2.0-py3-none-any.whl	200	fetch	pyodide.asm.js:10	9.8 kB	459 ms	■
☐	certifi-2022.12.7-py3-none-any.whl	200	fetch	pyodide.asm.js:10	156 kB	648 ms	■

40 requests | 43.3 MB transferred | 52.6 MB resources | Finish: 4.79 s | DOMContentLoaded: 127 ms | Load: 189 ms

Obrázek 21 - Obrázek z vývojářského prostředí ze sekce Network II

Samotný PyScript je distribuován v jednom JS souboru čítající zhruba třicet tisíc řádků kódu. Takto si můžeme udělat obrázek o složitosti samotného frameworku, jež zahrnuje celou řadu dalších nástrojů.

6.1.1 Soubory s příponou .whl

Během práce se můžeme setkat se soubory s příponou .whl (Wheel). Formát .whl je používán pro distribuci balíčků a je jedním z formátů, který je podporován nástrojem pro správu balíčků pip.

Soubory .whl jsou binární distribuce balíčků obsahující předkompilované soubory a metadata potřebná pro jejich instalaci. Tyto soubory jsou optimalizovány pro konkrétní platformu, což znamená, že mohou obsahovat již přeložený kód, což zrychluje proces instalace.

Při instalaci balíčku pomocí nástroje pip můžete specifikovat, zda chcete nainstalovat balíček přímo ze zdrojového kódu (např. ze souboru s příponou .tar.gz nebo .zip) nebo z předkompilovaného souboru .whl. Z obrázků je patrné, že PyScript kombinuje obě možnosti. Výhodou instalace z předkompilovaného souboru .whl je rychlejší proces instalace a snadnější řešení závislostí.

Přestože soubory .whl mohou být větší než zdrojové kódy, představují efektivní způsob distribuce a instalace balíčků Pythonu, zejména pokud je třeba optimalizovat výkon a minimalizovat závislosti. [68]

6.2 Významné balíčky

Mezi významné a široce používané balíčky jazyku Python patří několik nástrojů, jež využívá i framework PyScript. Jejich kombinací můžeme dosáhnout velmi silného ekosystému, který

je pro uživatele snadno použitelný a přívětivý. Níže nalezneme krátký popis a zaměření jednotlivých balíčků a jejich přínosů.

6.2.1 NTLK

NLTK (Natural Language Toolkit) je rozsáhlý balíček pro přírodní jazykové zpracování (NLP) v jazyce Python. Jeho cílem je poskytnout vývojářům a výzkumníkům nástroje a prostředky pro práci s textem a jeho analýzou. NLTK nabízí mnoho funkcí a algoritmů pro zpracování, manipulaci a modelování textových dat.

Zde je stručný přehled některých klíčových vlastností a funkcí, které NLTK poskytuje:

- **Tokenizace:** NLTK nabízí různé algoritmy pro rozdělení textu na tokeny (jednotlivá slova nebo fráze). Tokenizace je často prvním krokem při zpracování textu.
- **Stop Words:** Balíček obsahuje seznam často se vyskytujících slov, které jsou obvykle bezvýznamné pro analýzu textu (např. spojky, předložky). NLTK poskytuje nástroje pro jejich identifikaci a odstranění.
- **Stemming a Lemmatizace:** NLTK podporuje algoritmy pro redukci slov na jejich základní tvar (stemming) a pro nalezení jejich základního slovního tvaru (lemmatizace). Tyto techniky pomáhají zjednodušit analýzu textu a zlepšit shodu mezi slovy.
- **Početní a statistické metody:** NLTK obsahuje různé algoritmy pro analýzu frekvence slov, klasifikaci textu, extrakci klíčových slov a další statistické metody pro zpracování textu.
- **Syntaktická analýza:** Balíček poskytuje nástroje pro syntaktickou analýzu větné struktury, jako je rozpoznávání frází, závislostí mezi slovy a analýza stromů.
- **Korpuse a lexikony:** NLTK obsahuje rozsáhlou kolekci textových korpusů (sbírek textů) a lexikonů, které mohou být použity pro trénování modelů a testování algoritmů.

NLTK je oblíbeným nástrojem v akademickém prostředí a ve výzkumných oblastech spojených s přírodním jazykovým zpracováním. Je otevřeným zdrojem a je aktivně udržován komunitou, což znamená, že je pravidelně aktualizován a rozšiřován o nové funkce a algoritmy.

NLTK je mocný a rozsáhlý balíček pro přírodní jazykové zpracování v jazyce Python, který poskytuje širokou škálu nástrojů a funkcí pro analýzu a manipulaci s textovými daty. Jeho snadné použití, bohatá dokumentace a aktivní komunita ho činí populárním nástrojem pro všechny, kteří se zajímají o práci s textem a jazykem v prostředí Python. [69]

6.2.2 Pandas

Pandas je výkonný a oblíbený balíček pro jazyk Python, který se používá pro práci s daty. Je navržen tak, aby usnadňoval manipulaci, analýzu a zpracování datových struktur, jako jsou tabulky a časové řady. Pandas poskytuje vysokoúrovňové funkce a nástroje, které umožňují efektivní a flexibilní práci s daty.

Hlavními komponentami balíčku Pandas jsou dvě základní datové struktury: DataFrame a Series. DataFrame je dvourozměrná tabulka, která obsahuje řádky a sloupce, zatímco Series je jednorozměrná struktura, která představuje jednu sloupcovou nebo řádkovou sadu dat. Tyto datové struktury umožňují snadnou manipulaci, filtrování, sloučení, agregaci a analyzování dat.

Pandas nabízí mnoho funkcí pro načítání a ukládání dat z různých formátů, včetně CSV, Excel, SQL databází a dalších. Navíc poskytuje pokročilé funkce pro práci s chybějícími daty, indexování, vybírání a filtrování dat, transformaci dat, řazení, slévání a spojování datových sad, statistické výpočty, vizualizaci dat a mnoho dalších.

Díky svému silnému nástrojovému arzenálu a snadnému použití se Pandas stal preferovanou volbou pro analytiky, datové vědce a vývojáře, kteří pracují s daty. Jeho funkce umožňují rychlé prototypování, čištění a transformaci dat, což je zásadní pro přípravu dat pro modelování a analýzu.

Pandas je často používán ve spojení s dalšími balíčky pro datovou analýzu a strojové učení, jako je například NumPy, Matplotlib, SciPy a Scikit-learn. Tyto balíčky spolu vytvářejí silný ekosystém pro práci s daty a umožňují provádět komplexní analýzy, vizualizace a modelování. [70]

6.2.3 Pillow

Pillow je balíček pro jazyk Python, který slouží k práci s obrázky. Jedná se o moderní a jednoduchý nástroj, který poskytuje rozhraní pro manipulaci s různými formáty obrázků, včetně JPEG, PNG, GIF, BMP a mnoha dalších.

Hlavním cílem balíčku Pillow je umožnit vývojářům snadno načítat, manipulovat a ukládat obrázky pomocí Pythonu. Poskytuje širokou škálu funkcí pro úpravu obrázků, jako je změna velikosti, ořezání, rotace, zrcadlení, změna barev, přidávání textu a mnoho dalšího.

Pillow je navržen tak, aby byl jednoduchý na použití a zároveň dostatečně výkonný pro zpracování velkých a složitých obrázků. Poskytuje přehledné a intuitivní rozhraní, které umožňuje snadnou integraci s existujícími projekty a workflow.

Je jedním z hlavních balíčků pro práci s obrázky v Pythonu a často se používá ve spojení s dalšími balíčky pro vizualizaci dat, strojové učení a zpracování obrazu.

Díky své flexibilitě a podpoře různých formátů obrázků se Pillow hodí pro různé účely, včetně webových aplikací, vědeckého výzkumu, tvorby grafiky, zpracování obrazu a mnoho dalších. Jeho snadné použití a bohatá dokumentace umožňují vývojářům rychle a efektivně pracovat s obrázky pomocí Pythonu. [71]

6.2.4 Bokeh

Bokeh je interaktivní vizualizační knihovna pro jazyk Python, která umožňuje vývojářům vytvářet bohaté a interaktivní grafy, diagramy a vizualizace dat pro webové stránky nebo aplikace. Je navržena tak, aby umožnila snadné a efektivní zobrazení dat ve formě atraktivních a interaktivních grafů.

Hlavním cílem Bokeh je poskytnout uživatelům prostředky pro rychlé a snadné vytváření vizualizací dat, které mohou být interaktivně ovládány pomocí webového rozhraní. Knihovna podporuje širokou škálu grafických prvků, včetně čar, sloupcových a spojnicových grafů, histogramů, rozptylových diagramů, plošných grafů a mnoho dalšího. Umožňuje také přidávání interaktivních prvků, jako jsou nástroje pro přibližování, pohyb, výběr a další.

Bokeh je postaven na moderních webových technologiích, jako je HTML, CSS a JavaScript, a využívá moderních prohlížečů k vykreslování grafů. To znamená, že výsledné vizualizace mohou být snadno sdíleny a distribuovány prostřednictvím webových stránek nebo nasazeny jako součást webových aplikací.

Jedním z klíčových prvků je jeho schopnost generovat interaktivní vizualizace pomocí Pythonu. Uživatelé mohou vytvářet grafy a ovládat jejich chování prostřednictvím Python

API. Umožňuje například propojení vizualizací s daty, takže při změně dat se automaticky aktualizují i zobrazené grafy. To poskytuje dynamické a interaktivní zážitky pro uživatele.

Nabízí rozšiřitelnost a flexibilitu prostřednictvím podpory rozšíření a rozhraní s dalšími populárními knihovnamy pro analýzu dat v Pythonu, jako je NumPy a Pandas. To umožňuje snadnou integraci Bokeh s existujícími projekty a zdroji dat. [72]

6.2.5 Scipy

Scipy je knihovna pro programovací jazyk Python, která poskytuje funkce a nástroje pro vědecké výpočty a numerickou analýzu. Je navržena tak, aby usnadňovala manipulaci s různými typy dat, numerické výpočty, optimalizaci, statistiku, lineární algebru a další matematické operace.

Scipy staví na jiné populární knihovně pro výpočty, kterou je NumPy. NumPy rozšiřuje základní funkcionality Pythonu o multidimenzionální pole a efektivní operace nad nimi. Scipy na to navazuje a přidává další vyšší úrovně funkce, které jsou často využívány ve vědeckých disciplínách a inženýrských aplikacích. Staví také na balíčcích Matplotlib a Pandas.

Scipy nabízí širokou škálu modulů a funkcí. Mezi hlavní patří:

- Modul `scipy.integrate`: Poskytuje funkce pro numerickou integraci diferenciálních rovnic, integraci funkcí a kvadraturu.
- Modul `scipy.optimize`: Obsahuje optimalizační algoritmy pro nalezení extrémů funkcí, včetně hledání minima nebo maxima, globální optimalizace a vyhledávání kořenů rovnic.
- Modul `scipy.stats`: Poskytuje širokou škálu pravděpodobnostních distribucí, statistických funkcí a testů pro analýzu dat a pravděpodobnostních modelů.
- Modul `scipy.linalg`: Obsahuje funkce pro lineární algebru, jako je řešení lineárních rovnic, výpočet determinantu a inverze matice.
- Modul `scipy.signal`: Poskytuje nástroje pro analýzu a manipulaci s signály a systémy, včetně filtrů, spektrální analýzy a konvoluce.
- Modul `scipy.interpolate`: Slouží k interpolaci dat a vytváření hladkých křivek z naměřených hodnot.

- Modul `scipy.spatial`: Obsahuje algoritmy pro výpočet vzdálenosti mezi body, konstrukci Delaunayho triangulace a další operace s geometrickými daty.

Scipy je široce využívána ve vědecké komunitě a průmyslu pro numerické výpočty, modelování a analýzu dat. Její síla spočívá v tom, že poskytuje efektivní a optimalizované implementace algoritmů, které jsou důležité pro vědecké výpočty, a to vše v jednoduchém a přístupném rozhraní jazyka Python. [73]

6.2.6 Scikit-learn

Scikit-learn je open-source knihovna pro strojové učení, která je vysoce populární mezi vývojáři a datovými vědci v jazyce Python. Poskytuje širokou škálu algoritmů pro klasifikaci, regresi, shlukování, redukci dimenzionality, předzpracování dat a mnoho dalších úloh spojených se strojovým učáním.

Scikit-learn je vhodná pro začátečníky i pokročilé uživatele, kteří se zabývají strojovým učáním a potřebují snadno použitelný nástroj s širokou škálou algoritmů a funkcí. Je navržena s důrazem na efektivitu, robustnost a jednoduchost použití, což z ní činí jednu z nejoblíbenějších knihoven pro strojové učení v jazyce Python.

Vzhledem k tomu, že Scikit-learn je zdarma dostupný a licencován pod licencí BSD, umožňuje svobodné používání a šíření bez omezení. To umožňuje vývojářům, vědcům a studentům využívat tuto knihovnu pro své projekty, výzkum a vzdělávání.

Scikit-learn je mocný a všestranný nástroj pro strojové učení v jazyce Python. S jeho pomocí můžete snadno vytvářet a trénovat modely pro různé úlohy a aplikace. Jeho široká podpora algoritmů, jednoduché rozhraní a dobře zdokumentované funkce ho činí neocenitelným nástrojem pro každého, kdo se zajímá o strojové učení a analytiku dat. [74]

6.2.7 Beautiful Soup

Beautiful Soup je balíček pro Python, který slouží k extrakci dat z webových stránek. Je navržen tak, aby usnadňoval zpracování a analýzu HTML a XML dokumentů, a umožňuje snadný přístup k jejich jednotlivým elementům.

Jedna z hlavních vlastností Beautiful Soup je jeho schopnost zpracovat i nevalidní nebo špatně formátovaný HTML kód. Může se tak vypořádat s různými chybami, které by jinak mohly způsobit potíže při parsování dat.

S pomocí Beautiful Soup můžete provádět různé operace nad webovými stránkami. Například můžete vyhledávat určité elementy na stránce pomocí selektorů (CSS nebo XPath), získávat hodnoty atributů, procházet stromovou strukturu dokumentu a extrahovat požadovaná data.

Pro svou jednoduchou a intuitivní syntaxi je Beautiful Soup oblíbeným nástrojem mezi vývojáři a datovými analytiky. Je snadno použitelný pro různé úlohy, jako je web scraping, extrakce informací, analýza dat, automatizace a další.

Beautiful Soup je dostupný jako open-source projekt s otevřenou licencí, což umožňuje jeho volné používání a úpravy. Má také aktivní komunitu, která poskytuje podporu, aktualizace a rozšiřování funkcí.

Jde o užitečný nástroj pro zpracování a extrakci dat z webových stránek v jazyce Python. Jeho flexibilita, schopnost zpracovávat špatně formátovaný HTML kód a snadné použití činí z Beautiful Soup nezbytný prvek v arzenálu každého, kdo se zabývá web scrapingem, analýzou dat nebo automatizací webu. [75]

6.2.8 Numpy

Numpy je balíček pro Python, který přináší podporu pro efektivní práci s numerickými daty. Je jedním z nejpopulárnějších a nejdůležitějších balíčků v oblasti vědeckého výpočetního prostředí a datové analýzy.

Hlavním cílem balíčku Numpy je umožnit efektivní manipulaci s vícerozměrnými poli (ndarrays). Tyto pole jsou rychlé, úsporné při paměti a umožňují provádět různé matematické operace. Numpy poskytuje mnoho funkcí a metod pro práci s těmito poli, jako jsou operace pro indexování, řezání (slicing), přetváření, spojování, rozdělování a mnoho dalších.

Další klíčovou vlastností balíčku Numpy je podpora rozsáhlého množství matematických operací. Obsahuje základní matematické funkce (jako jsou sin, cos, exp, log) a umožňuje provádět operace jako násobení, dělení, sčítání a odečítání nad celými poli dat. Díky tomu je Numpy často používán pro numerické výpočty, lineární algebru, statistiku a další matematické úlohy.

Numpy také poskytuje mnoho dalších užitečných nástrojů a funkcionalit, jako je generování náhodných čísel, integrace s jinými balíčky (např. Matplotlib pro vizualizaci dat), lineární algebra, Fourierova transformace, práce s polynomy a mnoho dalšího.

Díky své optimalizované implementaci v jazyce C a C++ je Numpy velmi rychlý a efektivní. Je navržen tak, aby umožňoval práci s velkými objemy dat a zároveň minimalizoval spotřebu paměti. Díky tomu je vhodný pro zpracování rozsáhlých datových sad a numerické výpočty ve vědeckých a datových analýzách.

Numpy je široce používán v různých oblastech, jako jsou strojové učení, analýza dat, fyzika, biologie, ekonomie a další. Díky své robustnosti, výkonu a rozsáhlým funkcím je Numpy nepostradatelným nástrojem pro každého vědce, datového analytika nebo vývojáře, který se zabývá numerickými výpočty a manipulací s daty v Pythonu. [76]

6.2.9 Matplotlib

Matplotlib je balíček pro vizualizaci dat v programovacím jazyce Python. Poskytuje širokou škálu funkcí a nástrojů pro tvorbu různých druhů grafů, diagramů a vizualizací. Je jedním z nejpopulárnějších nástrojů pro vizualizaci dat v Pythonu a je často používán ve vědeckém výzkumu, datové analýze, strojovém učení a mnoha dalších oblastech.

Matplotlib je navržen tak, aby byl jednoduchý na použití, ale zároveň dostatečně mocný pro pokročilé vizualizační úlohy. Podporuje různé typy grafů, jako jsou čárové grafy, sloupcové grafy, bodové grafy, koláčové grafy, histogramy, plošné grafy, 3D grafy a mnoho dalších. Uživatelé mohou přizpůsobit vzhled grafů a diagramů pomocí různých konfiguračních možností, jako jsou barvy, styly čar, popisky os, mřížky a mnoho dalšího.

Matplotlib je postaven na vektorové grafické knihovně, která umožňuje vytvářet vysokokvalitní grafiku, která je vhodná pro tisk, prezentace a online publikace. Grafy vytvořené pomocí Matplotlibu mohou být exportovány do různých formátů, včetně PNG, PDF, SVG a dalších.

Balíček Matplotlib je velmi dobře dokumentovaný a má rozsáhlou a aktivní komunitu uživatelů. To znamená, že je snadné najít podporu, příklady kódu a řešení pro různé vizualizační úlohy. Existuje také mnoho rozšíření a nadstaveb, které rozšiřují funkčnost Matplotlibu a poskytují ještě více možností pro vizualizaci dat.

Díky své jednoduchosti, rozsáhlým funkcím a flexibilitě je Matplotlib nezbytným nástrojem pro každého, kdo se zabývá vizualizací dat v Pythonu. Bez ohledu na to, zda potřebujete vytvořit jednoduchý graf nebo složitou vizualizaci, Matplotlib poskytuje veškeré nástroje, které potřebujete k prezentaci a interpretaci dat ve svém projektu. [77]

ZÁVĚR

PyScript je framework, který se stále aktivně vyvíjí, a v budoucnu je očekávána zlepšená funkcionalita. Tento fakt je důležitým aspektem, který je třeba vzít v úvahu při posuzování tohoto frameworku. Aktualizace a vývoj mohou přinášet nové funkce a vylepšení, které mohou zvýšit efektivitu a pohodlí vývojářů při práci s PyScriptem. Je tedy možné, že v budoucnu budou knihovny stále optimalizovány a redukovány, aby se minimalizoval objem dat, který je nutné stahovat.

Důležité je si však také uvědomit, že v současné době je dokumentace k PyScriptu často nepřehledná a méně obsáhlá. Přístupná dokumentace je nezbytná pro vývojáře, aby se seznámili s funkcemi a možnostmi frameworku a aby se mohli s ním efektivněji pracovat. V současnosti může být obtížné najít kompletní a srozumitelnou dokumentaci k PyScriptu, což může omezovat jeho širší přijetí a popularitu. Faktorem, který je třeba zohlednit, je velikost komunity kolem PyScriptu. Aktivní a rozsáhlá komunita je důležitá pro podporu, výměnu informací a řešení problémů, které se mohou při vývoji objevit.

Práce s PyScriptem zatím neposkytuje uspokojivý výsledek, protože zmíněné problémy ztěžují vývojářům práci a brání jim v plném prozkoumání možností tvorby webových stránek v jazyce Python. V porovnání s jinými frameworky, které se zaměřují na stejnou problematiku, PyScript zůstává v některých ohledech pozadu.

Je důležité zdůraznit, že PyScript se vydává trochu odlišnou cestou oproti ostatním frameworkům. Jeho zaměření na využití jazyka Python a přenos této síly do vývoje webových aplikací je inovativní a slibné. Avšak je zapotřebí vyčkat na pokročilejší fáze jeho vývoje, aby se překonaly současné problémy a framework byl schopen plně konkurovat a poskytovat lepší řešení. Zatímco jiné frameworky v této oblasti nabízejí více pokročilé funkce, robustnější knihovny a rozsáhlejší dokumentaci, PyScript ještě nedosáhl podobné úrovně.

Na závěr lze říci, že PyScript je perspektivní framework, který nabízí výhody psaní kódu v jazyce Python. Avšak je důležité mít na paměti, že je stále ve vývoji a zatím se potýká s některými nedostatky, jako jsou velké knihovny, nepřehledná dokumentace a omezená komunita. Při rozhodování o použití PyScriptu je proto nezbytné zvážit všechny tyto faktory a vyhodnotit, zda framework odpovídá specifickým potřebám a požadavkům projektu.

SEZNAM POUŽITÉ LITERATURY

- [1] Stack Overflow Developer Survey 2022. Stack Overflow [online]. [cit. 2023-05-22]. Dostupné z: <https://survey.stackoverflow.co/2022/#most-popular-technologies-language>
- [2] GRULLON, Manuel. Blurred Lines: Is Ruby an interpreted language and what does that even mean?. Medium [online]. 2018 [cit. 2023-05-22]. Dostupné z: <https://medium.com/@astermanuelg/blurred-lines-is-ruby-an-interpreted-language-2d3d6bca3d37>
- [3] A tour of the C# language. Microsoft Learn [online]. 2023 [cit. 2023-05-22]. Dostupné z: <https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>
- [4] What Is The Difference Between A .NET Developer And An ASP.NET Developer?. Cisin [online]. [cit. 2023-05-22]. Dostupné z: <https://www.cisin.com/coffee-break/technology/what-is-the-difference-between-a-net-developer-and-an-asp-net-developer.html>
- [5] TOUB, Stephen. Performance Improvements in .NET 7. .NET Blog [online]. 2022 [cit. 2023-05-22]. Dostupné z: https://devblogs.microsoft.com/dotnet/performance_improvements_in_net_7/
- [6] LOCK, Andrew. Understanding the .NET ecosystem: The evolution of .NET into .NET 7. Andrew Lock .NET Escapades [online]. 2023 [cit. 2023-05-19]. Dostupné z: <https://andrewlock.net/understanding-the-dotnet-ecosystem-the-evolution-of-dotnet-into-dotnet-7/>
- [7] PHP vs. Python – the two programming languages compared. IONOS [online]. 2023 [cit. 2023-05-22]. Dostupné z: <https://www.ionos.com/digitalguide/websites/web-development/php-vs-python/>
- [8] KHANA, Naveen. How Does PHP Compare to Other Programming Languages?. EBizNeeds [online]. [cit. 2023-05-22]. Dostupné z: <https://www.ebizneeds.com/blog/how-does-php-compare-to-other-programming-languages/>
- [9] BROTHERTON, Claire. PHP vs JavaScript: An In-Depth Comparison of the Two Scripting Languages. Kinsta [online]. 2022 [cit. 2023-05-22]. Dostupné z: <https://kinsta.com/blog/php-vs-javascript/>

- [10] ACHARYA, Durga Prasad. PHP vs Python: A Detailed Comparison Between the Two Languages. Kinsta [online]. 2023 [cit. 2023-05-22]. Dostupné z: <https://kinsta.com/blog/php-vs-python/>
- [11] GOEL, Aman. Python vs PHP in 2023. Hackr.io [online]. 2023 [cit. 2023-05-22]. Dostupné z: <https://hackr.io/blog/python-vs-php>
- [12] PHP vs Python in 2023: An In-depth Comparison. DOIT Software [online]. 2023 [cit. 2023-05-22]. Dostupné z: <https://doit.software/blog/php-vs-python>
- [13] The Good and the Bad of Java Programming. Altexsoft [online]. 2018 [cit. 2023-05-22]. Dostupné z: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-java-programming/>
- [14] KRAVCHENKO, Iryna. Pros and cons of Java development for your project. Diceus [online]. 2022 [cit. 2023-05-22]. Dostupné z: <https://diceus.com/why-is-java-so-popular/>.
- [15] ADVANTAGES AND DISADVANTAGES OF JAVA. Boot Poot [online]. 2021 [cit. 2023-05-22]. Dostupné z: <https://bootpoot.tech/advantages-and-disadvantages-of-java/>
- [16] Advantages and Disadvantages of Java. Tech Vidvan [online]. [cit. 2023-05-22]. Dostupné z: <https://techvidvan.com/tutorials/pros-and-cons-of-java/>
- [17] Kotlin docs. Kotlin [online]. [cit. 2023-05-23]. Dostupné z: <https://kotlinlang.org/docs/home.html>
- [18] RATHORE, Deven. 10 best kotlin web frameworks. Dunebook [online]. 2019 [cit. 2023-05-23]. Dostupné z: <https://www.dunebook.com/best-kotlin-web-frameworks/>
- [19] SILVERIO, Manuel. What is Kotlin?. Builtin [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://builtin.com/software-engineering-perspectives/kotlin>
- [20] Advantages and Disadvantages of Kotlin. Krify [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://krify.co/advantages-and-disadvantages-of-kotlin/>
- [21] The Advantages and Disadvantages of JavaScript. FreeCodeCamp [online]. 2019 [cit. 2023-05-23]. Dostupné z: <https://www.freecodecamp.org/news/the-advantages-and-disadvantages-of-javascript/>
- [22] JavaScript. Mozilla Developer [online]. [cit. 2023-05-23]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/javascript>

- [23] A., Jordana. What Is JavaScript? A Basic Introduction to JS for Beginners. Hostinger [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://www.hostinger.com/tutorials/what-is-javascript>
- [24] SIMRAN KAUR, Arora. 10 Best JavaScript Frameworks to Use in 2023. Hackr.io [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://hackr.io/blog/best-javascript-frameworks>
- [25] 17 JavaScript Frameworks that You Should Know About – A Comprehensive Guide. Codemotion [online]. 2021 [cit. 2023-05-23]. Dostupné z: <https://www.codemotion.com/magazine/frontend/javascript/javascript-frameworks-guide/>
- [26] MOUSKHELICHVILI, Tim. TypeScript vs JavaScript: Which One You Should Use, and Why. Sitepoint [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://www.sitepoint.com/typescript-vs-javascript/>
- [27] TypeScript Documentation. TypeScript Documentation [online]. [cit. 2023-05-23]. Dostupné z: <https://www.typescriptlang.org/docs/>
- [28] ABHIMANYU, Krishnan. TypeScript vs JavaScript: Which is Best in 2023. Hackr.io [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://hackr.io/blog/typescript-vs-javascript>
- [29] USMAN, Malik. Advantages and Disadvantages of the Python Programming Language. Learn Python [online]. 2019 [cit. 2023-05-23]. Dostupné z: <https://learnpython.com/blog/python-programming-advantages-disadvantages/>
- [30] Python vs. Java: Which Should I Learn?. Coursera [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://www.coursera.org/articles/python-vs-java>
- [31] The Good and the Bad of Python Programming Language. Altexsoft [online]. 2021 [cit. 2023-05-23]. Dostupné z: <https://www.altexsoft.com/blog/python-pros-and-cons/>
- [32] OLUSHEYE, Ifihanagbara. How to Use PyScript – A Python Frontend Framework. FreeCodeCamp [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://www.freecodecamp.org/news/pyscript-python-front-end-framework/>
- [33] PLIGER, Fabio. New from Anaconda: Python in the Browser. Anaconda [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://www.anaconda.com/blog/pyscript-python-in-the-browser>

- [34] PAALVAST, Leonieke. PyScript, a first exploration. Pybites [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://pybit.es/articles/pyscript-a-first-exploration/>
- [35] JOY, Ashwin. Pyscript Tutorial With Simple Code Examples. Pythonista Planet [online]. [cit. 2023-05-23]. Dostupné z: <https://pythonistaplanet.com/pyscript/>
- [36] TADRISI, Amir. Running Python in the Browser with WebAssembly. Testdriven.io [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://testdriven.io/blog/python-webassembly/>
- [37] ROBINSON, Muiru. PYSCRIPT - PYTHON FRONTEND FRAMEWORK. TopCoder [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://www.topcoder.com/thrive/articles/pyscript-python-frontend-framework>
- [38] ZACZYŃSKI, Bartosz. A First Look at PyScript: Python in the Web Browser. Real Python [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://realpython.com/pyscript-python-in-browser/>
- [39] PyScript Tutorial | Run Python Script in the Web Browser. Javatpoint [online]. [cit. 2023-05-23]. Dostupné z: <https://www.javatpoint.com/pyscript>
- [40] YANG, Sophia. PyScript: Python in the Browser. Anaconda [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://anaconda.cloud/pyscript-python-in-the-browser>
- [41] WebAssembly. Mozilla Developer [online]. [cit. 2023-05-23]. Dostupné z: <https://developer.mozilla.org/en-US/docs/WebAssembly>
- [42] HERNANDEZ, Luis. What Is WebAssembly And What Is It Used For?. Fullstack Labs [online]. [cit. 2023-05-23]. Dostupné z: <https://www.fullstacklabs.co/blog/what-is-webassembly>
- [43] DECLUTE, Darcy. Why WebAssembly? Top 11 Wasm benefits. The Server Side [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://www.theserverside.com/tip/Why-WebAssembly-Top-Wasm-benefits>
- [44] Pyodide. Pyodide [online]. [cit. 2023-05-23]. Dostupné z: <https://pyodide.org/en/stable/>
- [45] BOYD, Eleanor. Feasibility, Use Cases, and Limitations of Pyodide. Microsoft Dev Blogs [online]. 2021 [cit. 2023-05-23]. Dostupné z: <https://devblogs.microsoft.com/python/feasibility-use-cases-and-limitations-of-pyodide/>
- [46] Emscripten. Emscripten [online]. [cit. 2023-05-23]. Dostupné z: <https://emscripten.org/>

- [47] ZAKAI, Alon. Emscripten: an LLVM-to-JavaScript compiler [online]. 2011 [cit. 2023-05-23]. Dostupné z: doi:10.1145/2048147.2048224
- [48] AHUJA, Abhisar. The Supreme Guide To Understand The Workings Of CPython. Simplilearn [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://www.simplilearn.com/tutorials/python-tutorial/understand-the-workings-of-cpython>
- [49] SHAW, Anthony. Your Guide to the CPython Source Code. RealPython [online]. [cit. 2023-05-23]. Dostupné z: <https://realpython.com/cpython-source-code-guide/>
- [50] Brython. Brython [online]. [cit. 2023-05-23]. Dostupné z: <https://brython.info/index.html>
- [51] BURGAUD, Andre. Brython: Python in Your Browser. RealPython [online]. [cit. 2023-05-23]. Dostupné z: <https://realpython.com/brython-python-in-browser/>
- [52] What is Transcrypt. Transcrypt [online]. [cit. 2023-05-23]. Dostupné z: https://www.transcrypt.org/docs/html/what_why.html#what-is-transcrypt
- [53] DE HOOGE, Jacques. Transcrypt: Anatomy of a Python to JavaScript Compiler. Infoq [online]. 2017 [cit. 2023-05-23]. Dostupné z: <https://www.infoq.com/articles/transcrypt-python-javascript-compiler/>
- [54] Skulpt. Skulpt [online]. [cit. 2023-05-23]. Dostupné z: <https://skulpt.org/>
- [55] Skulpt. GrumpyCoder [online]. [cit. 2023-05-23]. Dostupné z: <https://static.grumpycoder.net/pixel/skulpt/doc/>
- [56] FULTON, Aaron. Pywebview — The better alternative to Electron for Python programmers. Medium [online]. 2023 [cit. 2023-05-23]. Dostupné z: <https://medium.com/@aaronfulton/pywebview-the-better-alternative-to-electron-for-python-programmers-471d3c13693f>
- [57] KNIVETS, Max. PyWebView, a minimalistic Python library for rendering webpages (review). Max Knivets [online]. 2022 [cit. 2023-05-23]. Dostupné z: <https://maxknivets.com/pywebview/>
- [58] REACT TO PYTHON RESOURCES. PyReact [online]. [cit. 2023-05-23]. Dostupné z: <https://pyreact.com/resources#errata>
- [59] RapydScript. RapydScript Github [online]. [cit. 2023-05-23]. Dostupné z: <https://github.com/atsepkov/RapydScript#getting-started>

- [60] BeeWare. BeeWare [online]. [cit. 2023-05-24]. Dostupné z: <https://docs.beeware.org/en/latest/>
- [61] Welcome to Kivy. Kivy [online]. [cit. 2023-05-24]. Dostupné z: <https://kivy.org/doc/stable/>
- [62] Streamlit documentation. Streamlit [online]. [cit. 2023-05-24]. Dostupné z: <https://docs.streamlit.io/>
- [63] What is Pyjs. Pyjs [online]. [cit. 2023-05-24]. Dostupné z: <http://pyjs.org>
- [64] Jupyter Project Documentation. Jupyter Notebook [online]. [cit. 2023-05-24]. Dostupné z: <https://docs.jupyter.org/en/latest/>
- [65] Django documentation. Django [online]. [cit. 2023-05-24]. Dostupné z: <https://docs.djangoproject.com/en/4.2/>
- [66] What is HTML5 and what can I do with it?. Medium [online]. 2018 [cit. 2023-05-24]. Dostupné z: <https://medium.com/adalab/what-is-html5-and-what-can-i-do-with-it-d6bc85eb8af9>
- [67] Pip documentation. Pip [online]. [cit. 2023-05-24]. Dostupné z: <https://pip.pypa.io/en/stable/>
- [68] SHANKAR, Anjana. How to install a Python package with a .whl file. Educative.io [online]. [cit. 2023-05-24]. Dostupné z: <https://www.educative.io/answers/how-to-install-a-python-package-with-a-whl-file>
- [69] MOGYOROSI, Marius. Sentiment Analysis: First Steps With Python's NLTK Library. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/python-nltk-sentiment-analysis/>
- [70] HORVATH, Reka. Using pandas and Python to Explore Your Dataset. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/pandas-python-explore-dataset/>
- [71] GRUPETTA, Stephen. Image Processing With the Python Pillow Library – Real Python. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/image-processing-with-the-python-pillow-library/>

[72] D'ANGIO, Leon. Interactive Data Visualization in Python With Bokeh. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/python-data-visualization-bokeh/>

[73] WEBER, Bryan. Scientific Python: Using SciPy for Optimization. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/python-scipy-cluster-optimize/>

[74] STOJILJKOVIĆ, Mirko. Split Your Dataset With scikit-learn's train_test_split(). RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/train-test-split-python-data/>

[75] BREUSS, Martin. Beautiful Soup: Build a Web Scraper With Python. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/beautiful-soup-web-scraper-python/>

[76] PALO, Ryan. NumPy Tutorial: Your First Steps Into Data Science in Python. RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/numpy-tutorial/>

[77] SOLOMON, Brad. Python Plotting With Matplotlib (Guide). RealPython [online]. [cit. 2023-05-24]. Dostupné z: <https://realpython.com/python-matplotlib-guide/>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

API Application Interface

DOM Document Object Model

GUI Graphic User Interface

HTML Hyper Text Markup Language

JS JavaScript

NLP Natural Language Processing

OOP Object Oriented Programming

PDF Portable Document Format

TS TypeScript

XML Extensible Markup Language

SEZNAM OBRÁZKŮ

Obrázek 1 – Programovací, skriptovací a značkovací jazyky [1].....	12
Obrázek 2 – Databáze [1]	13
Obrázek 3 – Webové frameworky a technologie [1].....	14
Obrázek 4 – Ostatní frameworky a knihovny [1]	15
Obrázek 5 – Cloudové platformy [1].....	16
Obrázek 6 – Ostatní vývojové nástroje [1]	17
Obrázek 7 – Vývojová prostředí [1]	18
Obrázek 8 – Kompilované a interpretované jazyky [2].....	19
Obrázek 9 – Technologický přehled [40]	31
Obrázek 10 – Úvodní strana tutoriálu.....	58
Obrázek 11 – První strana tutoriálu	59
Obrázek 12 – Druhá strana tutoriálu.....	60
Obrázek 13 – Třetí strana tutoriálu	61
Obrázek 14 – Čtvrtá strana tutoriálu.....	62
Obrázek 15 – Pátá strana tutoriálu.....	62
Obrázek 16 – Šestá strana tutoriálu	63
Obrázek 17 – Sedmá strana tutoriálu.....	64
Obrázek 18 – První ukázka z editoru.....	65
Obrázek 19 – Druhá ukázka z editoru	65
Obrázek 20 – Obrázek z vývojářského prostředí ze sekce Network I.....	66
Obrázek 21 - Obrázek z vývojářského prostředí ze sekce Network II	67

SEZNAM PŘÍLOH

Příloha P I: Seznam příloh na cd

PŘÍLOHA P I: SEZNAM PŘÍLOH NA CD

fulltext.pdf – plný text diplomové práce

prilohy.zip – archiv obsahující zdrojové kódy aplikace