

Razvoj Programskih Jezika 2018

Seminarski rad u okviru kursa
Metodologija stručnog i naučnog rada
Matematički fakultet

Dušan Milović, Dušan Pilipović, Marko Petričić, Vido Mladenović

April 2019.

Sažetak

U ovom radu osvrnuli smo na razvoj i nastanak novih okvira, tehnologija, koncepta i programskih jezika u poslednjih nekoliko godina, a posebno smo prokomentarisali njihovo stanje u 2018. godini. Obradili smo trenutno stanje u industriji, prokomentarisali tehnologije koje koriste najveće kompanije i izdvojili ono za šta se očekuje da bude u žiži interesovanja i u narednom periodu.

Sadržaj

1 Uvod	3
2 Uloga industrije	3
3 Popularnost postojećih i nastanak novih jezika	9
4 Trendovi i noviteti	12
5 Dva novija jezika koja smo izdvojili	16
Literatura	22

1 Uvod

U današnje vreme život se ne može zamisliti bez modernih tehnologija. Potražnja za softverom je sve veća a broj ljudi koji imaju znanja da ga naprave nije tako veliki. Upravo ti ljudi koji razvijaju softvere se bave poslom koji je u poslednjih par godina jedan od najdinamičnijih poslova na svetu. Dinamika se ne ogleda u tome koliko treba fizički biti aktivan, već koliko je potrebno uložiti truda kako bi se ispratio razvoj svih tehnologija uz čiju pomoć razvijaju softvere. Upravo o tom razvoju ćemo pričati, o naglom skoku broja zainteresovanih za razvoj softvera kao i brzina razvoja tehnologija.

2 Uloga industrije

Bilo da se dvoumimo koji programski jezik da počnemo da učimo, ili u kojoj tehnologiji da započnemo novi projekat, veoma je važno da to što uradimo bude korisno i upotrebljivo i u narednom periodu. Programski jezik može da nastane, da se razvija i doživi ogromnu slavu i nakon toga da samo nestane i ne bude toliko popularan u svega godinu dana. Odabir tehnologije zavisi od prirode problema koji rešavamo ali i od ličnih zahteva i simpatija. Na mnogo načina možemo da poredimo programske jezike, ali doprinos boljoj odluci koju tehnologiju izabrati u velikoj meri može da predstavlja izučavanje šta je u centru pažnje u tehnološkoj industriji. Industrija će nam dati bitne trendove i znake koji će nam pomoći da donesemo odluku. Na primer ukoliko imamo informaciju da će određena tehnologija doneti najviše novca u 2018. ili da je najpopularnija u industriji, sigurno će nam skrenuti pažnju i povećati šanse da se usredsredimo baš na tu tehnologiju. U industriji je značajno više aspekata, pa ćemo se mi osvrnuti na tri najrelevantnija aspekta. Prvi o kojem ćemo pričati predstavlja količinu projekata u industriji koji su zasnovani na određenoj tehnologiji, zatim ćemo pomenuti tehnologije koje donose najlakšu zaposlenost i poslednje ali ne najmanje bitno koliko se isplati u finansijskom smislu koja tehnologija.

2.1 Projekti

Količina projekata u određenoj tehnologiji nam mnogo govori o toj tehnologiji i njenom razvoju. Iz toga možemo zaključiti pouzdanost tehnologije, njenu stabilnost, možemo shvatiti i koliko je isplativa ili neisplativa u zavisnosti od količine projekata. Loša strana posmatranja samo ovog pristupa može da bude to da u nekoj tehnologiji postoji veliki broj projekata, ali da je i veliki broj ljudi zna i koristi, pa možda potraga za poslom ne bude toliko jednostavan zadatak. I dalje ostaje otvoreno pitanje isplativosti te tehnologije jer možda ne zadovoljava lične zahteve i očekivanu dobit.

2.1.1 Uticaj velikih kompanija

Kao i u modi, tako i u svetu programiranja postoje nezvanični trendseteri u vidu velikih kompanija, čiji rad se neprestano prati i pokušavaju se izvući vrline koje čine te kompanije velikim. To su pre svega Microsoft, Facebook i Google, pa ćemo se osvrnuti na njihov rad i tehnologije koje najviše koriste. To nam predstavlja lep pokazatelj za tehnologiju da će verovatno biti veoma zastupljena u industriji.

Microsoft. Najpopularnija tehnologija koju razvijaju i koja je dosta primenjena je .NET. Ona podržava više objektno orijentisanih programskih jezika, C, F, Visual Basic. Za veb aplikacije pored .Net koriste se dosta Node.js. Za razvoj hibridnih aplikacija React Native i Cordova alati.

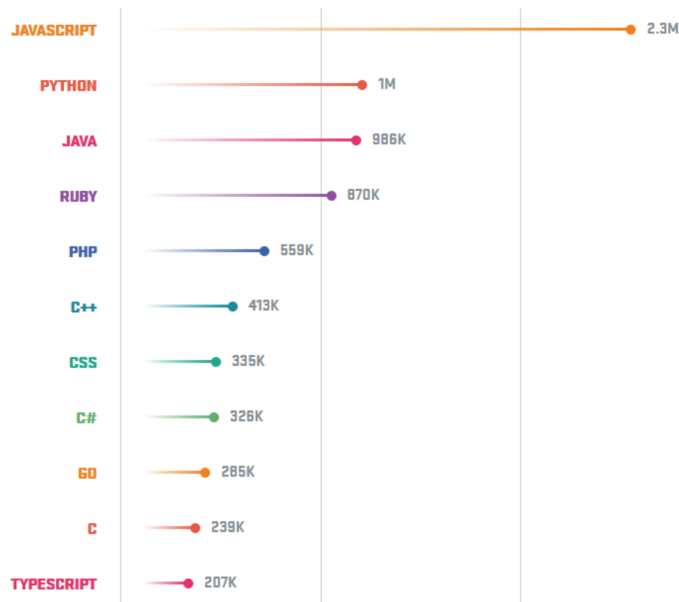
Facebook. Naravno prva asocijacija kada se pomene facebook medju programerima jeste React. Usled poteškoća oko održavanja svog koda za facebook sajt, rodila se ideja o nastajanju frameworka za Java Script koji je danas najpopularniji framework za veb programiranje.

Google kao jedna od vodećih IT kompanija ima veliki uticaj na razvoj jezika. Između ostalog, oni su oformili i svoj jezik pa ćemo njega navesti kao prvog na spisku korišćenog u Google-u, reč je o GO jeziku. Na front-end strani najviše koriste JavaScript, dok na back-end delu pored GO-a koriste još C, C++, Java, Python.

2.1.2 Analiza relevantnih sajtova

Pored tehnologija koje koriste velike kompanije kao gore navedene, bitno je da znamo i u kojim tehnologijama se rade projekti širom sveta. Dva veoma značajna sajta za razvoj softvera dostupni svim programerima su GitHub i Stackoverflow.

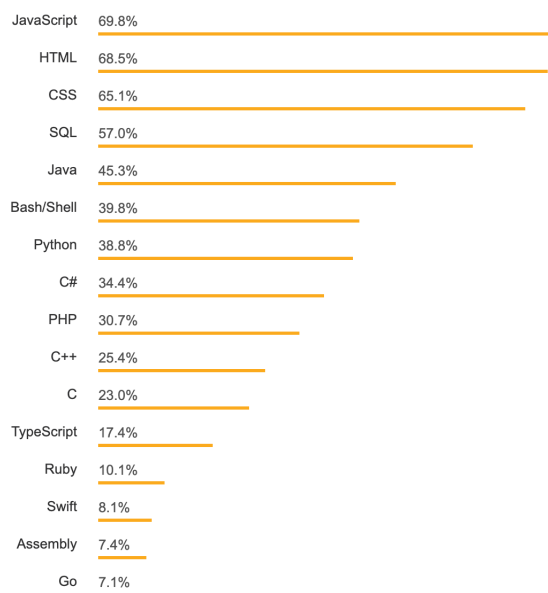
GitHub je sajt za deljenje koda, koji mnogo olakšava razvoj programa u timskom radu. Predstavlja najveću javnu bazu programa pa analize sprovedene na njemu imaju veliku težinu. Na osnovu evidencije GitHub-a možemo da vidimo na slici 1 koji jezik je najzastupljeniji kod korisnika nezavisno za kompanije u kojima rade.



Slika 1: Evidencija sa GitHub-a nezavisno od kompanije

Stackoverflow je sajt-forum namenjen programerima kako bi mogli da postavljaju pitanja i očekuju odgovore ljudi koji znaju da reše njihov problem. Jedna od najvećih zajednica programera, pa istraživanja sa

ovog sajta predstavljaju koliko toliko realnu sliku. Stackoverflow svake godine vrši anketu među korisnicima čije istraživanje ima za cilj da isprati trendove kao što su razvoj programskih jezika kroz godine, visina plate ljudi koji se bave određenom tehnologijom, količina posla za konkretan programski jezik. Na narednoj slici 2 možemo da vidimo rezultate za 2018. godinu najpopularnijih programskih jezika među ispitanicima sa sajta.



Slika 2: Evidencija sa Stack Overflow-a

2.1.3 Uticaj novih interesovanja

Veliki izazov u industriji predstavlja analiza velike količine podataka. Sve je više informacija i podataka i potrebno ih je analizirati, sortirati i izvući ono najbitnije iz njih. Za čoveka bi to bilo veoma teško da uradi, pa se teži tome da računar odradi ceo posao. Odatle se razvijaju teme kao što su "Big data", Mašinsko učenje, Veštačka inteligencija... Razvoj tih tema uslovljava i razvoj određenih programskih jezika. U svetu mašinskog učenja i obrade velike količine podataka najzastupljeniji su Python i R.

Python je doživio ogromnu porast u popularnosti tokom 2018. godine. Predočićemo vam neke prednosti Pythona.

- Python poseduje biblioteke koje su usmerene ka obradi podataka. To je jedini najveći razlog zašto se programeri sve više opredeljuju za Python. Nauka o podacima pruža zanimljiv posao ali je i veoma plaćen. Upoznaćemo se u kratkim crtama sa osnovnim bibliotekama Pythona namenjenih za oblast nauke o podacima:

Pandas: Programska biblioteka pisana za Python u svrhu analize i manipulacije podacima. Pruža pogodne strukture podataka i operacije za manipulisanje numeričkim tabelama i vremenskim serijama. Reč je o besplatnom programskom izdanju izdatom pod okriljem 3-klauzulne BSD licence.

NumPy: Ova biblioteka ima podršku za velike i višedimenzionalne nizove i matrice uporedo sa velikom kolekcijom matematičkih funkcija za operisanje nad ovim nizovima. Pandas je zasnovana na NumPy, i planirano je da se dobro uklapa sa ostalim naučnim kompjuterskim okruženjima tj nekim "3rd party" bibliotekama.

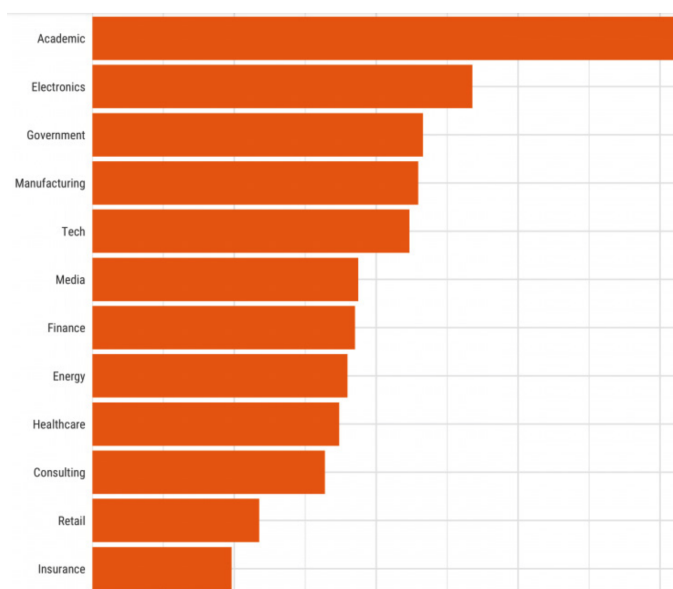
Matplotlib: To je Python 2D biblioteka za grafikone koja proizvodi podatke o kvalitetu u različitim formatima. Može da se koristi u Python skriptama, u Python i IPython shell, u Jupyter Notebook, u veb aplikacijama.

- Mašinsko učenje je oblast koja sve više uzima maha u IT industriji. Algoritmi u ovoj oblasti postaju sve prefinjeniji. Google-ova pretraga je najpribližniji primer kako unapred pogadja šta želimo da pretražujemo. Ovo je razlog da svako koga zanima mašinsko učenje i veštačka inteligencija treba da nauče Python koji je u ovoj oblasti trenutno najdominantniji jezik.

Gore navedene oblasti predstavljaju samo glavne prednosti učenja i rada u Python-u ali ne i jedine. Velika prednost Pythona jeste baš u tome što se ne koristi usko samo za jednu oblast, kao recimo R koji je dobar samo u analizi podataka i mašinskom učenju ali u veb programiranju nema svoju primenu, dok Python i u veb-u ima biblioteke kao što su Django i Flask.

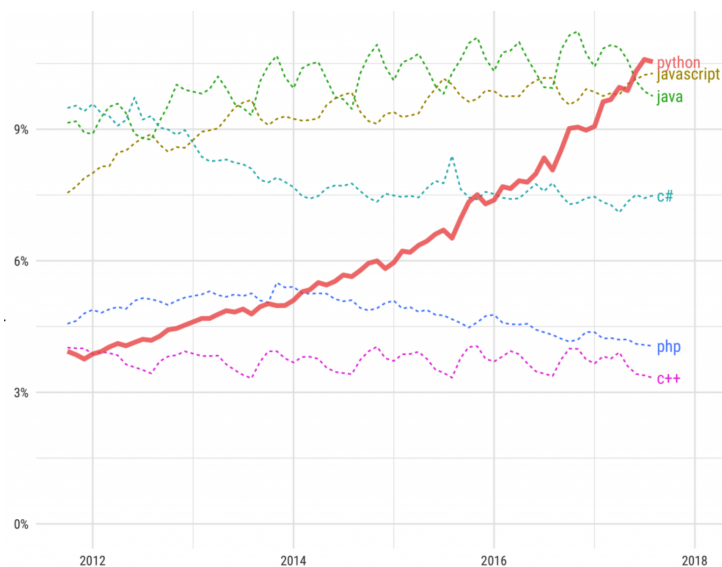
Učenje je dugačak proces, ali može se skratiti ukoliko imamo asistenta pri učenju. Uglavnom nam pomažu prijatelji, ali u slučaju da prijatelji nemaju odgovor na naše zahteve pomoć možemo potražiti i na internetu. Velika internet zajednica je velika prednost bilo kog jezika, a Python se može pohvaliti svojom širokom upotrebom i velikom zajednicom. To je veoma bitan faktor koji nam govori i o popularnosti jezika ali nam i olakšava učenje ovog jezika.

Prema anketama sa stackoverflow-a imamo sliku 3 na kojoj je prikazano u kojim industrijama se najviše koristi Python.



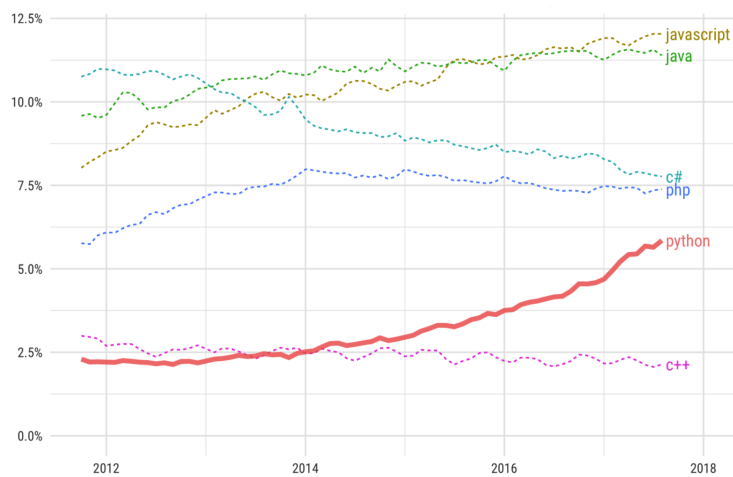
Slika 3: Evidencija sa Stack Overflow-a o upotrebi Pythona u industriji

Ukoliko uzmemo u obzir samo zemlje sa najvećim prihodima po svetskoj banci, razvoj najvećih programskih jezika po evidenciji na stackoverflow bi izgledao kao na narednoj slici 4.



Slika 4: Razvoj programskih jezika u ekonomski razvijenim zemljama

U poredjenju sa razvijenim zemljama po prihodima, na narednoj slici 5 možemo videti kako se kreće razvoj Pythona u ostalim zemljama, pa možda je to i razlog zašto nisu medju zemljama sa većim prihodima.

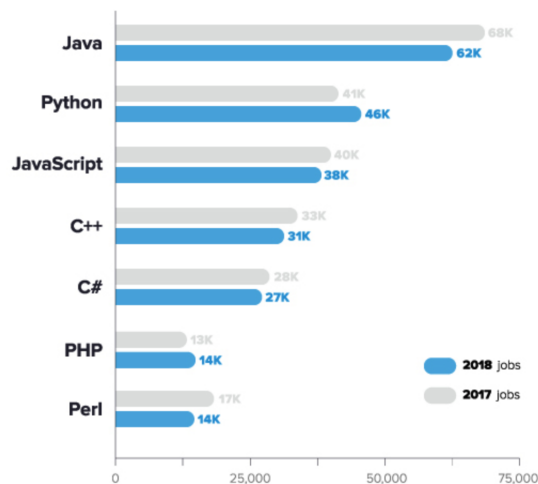


Slika 5: Razvoj programskih jezika u zemljama koje nisu u svetskom vrhu ekonomskog razvoja

Zaključak, Python je trenutno najbrže rastući jezik u oblasti analize podataka, a analiza podataka je trenutno najbrže rastuća oblast programiranja.

2.2 Zaposlenost

Popularnost programskih jezika se može na različite načine porediti. Smatramo da je najrelevantnija stopa zaposlenosti ljudi koji koriste određeni programski jezik. Da bismo prikazali stopu potražnje određenih programskih jezika u industriji, iskoristićemo analizu sprovedenu sa sajta Indeed.com (sajt za zapošljavanje) zasnovan na 25 programskih jezika, stack-ova, radnih okruženja. Analiza je zasnovana na broju ponuda za posao za svaki programski jezik. Primetićete da neki jezici se ne nalaze na listi top 7 iako su među programerima popularni, kao što su Ruby i Swift. Slika 6



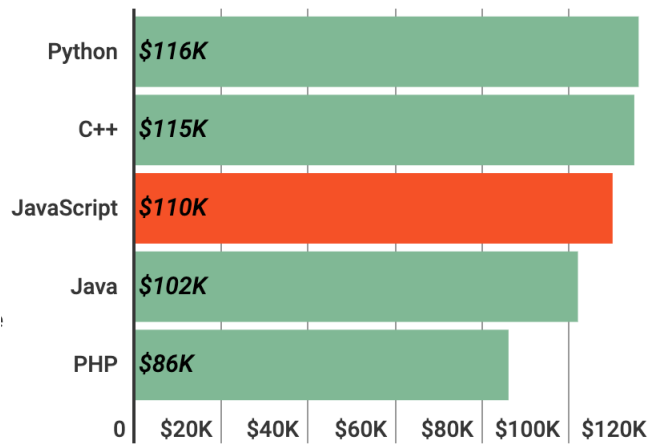
Slika 6: Nivo zaposlenosti po tehnologijama

2.3 Isplativost

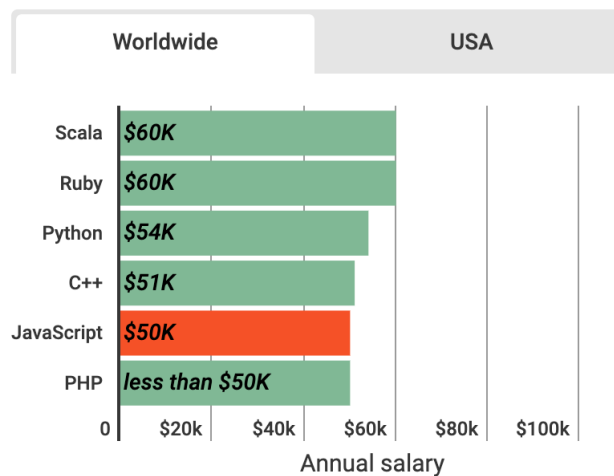
Za poslednju sekciju smo ostavili uticaj novca. Nije najmanje važna oblast, ali daleko i od toga da je najbitnija. Novac igra veliku ulogu u našim životima, međutim stanje se neće drastično promeniti u odnosu na izbor tehnologije. Ako smo nečemu posvećeni, možemo dostići velike uspehe u skoro svakoj tehnologiji. Po analizama sa raznih sajtova imamo okvirno koje tehnologije u proseku donose koliko novca. Naravno treba uzeti u obzir da se standard u različitim državama može drastično razlikovati, pa bukvalno poredjenje zarada nije adekvatno.

Prema sajtu Indeed.com koji smo već spominjali kao jedan od vodećih sajtova za zapošljavanje situacija u USA na godišnjem nivou je kao na slici 7.

Na osnovu podataka sa stackoverflow-a možemo uočiti da se situacija drastično menja ukoliko posmatramo količinu zarade širom sveta na godišnjem nivou, ne samo u USA. Slika 8.



Slika 7: Prosečna zarada u USA u odnosu na tehnologiju



Slika 8: Prosečna zarada u odnosu na tehnologiju širom sveta

3 Popularnost postojećih i nastanak novih jezika

Industrija programiranja se razvija pa tako i programski jezici. Istraživanja rađena na sajtu Github nam pokazuju u kojim programskim jezicima je rađeno najviše projekata, koji jezici su doživeli uspon, a koji pad u 2018. godini. Istraživanje je rađeno na osnovu mesečno aktivnih korisnika u proteklim godinama.

3.1 Najpopularniji jezici

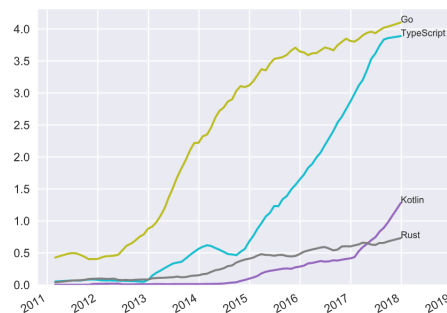
Rank	Language	Monthly Active Users	Trend
1	JavaScript	22.63%	
2	Python	14.75%	
3	Java	14.01%	
4	C++	8.45%	
5	C	6.03%	
6	PHP	5.85%	
7	C#	5.03%	
8	Shell	4.85%	
9	Go	4.10%	
10	TypeScript	3.89%	

Lista najpopularnijih programskih jezika nije se drastično promenila. Ovi jezici imaju relativno stabilnu upotrebu u proteklih par godina. JavaScript, Python, Java, C++ i C su najpopularniji jezici više od 8 godina i ne vidi se neka promena u skorije vreme.

JavaScript ostaje na prvom mestu najpopularnijih jezika uz pomoć brojnih framework-a koji su privukli još više korisnika. Python vremenom raste i nedavno je prestigao Javu i postao drugi najpopularniji programski jezik. Kao što smo već spomenuli uspehu Python-a doprineo je nagli porast interesovanja u oblasti mašinskog učenja.

3.2 Noviji jezici

Jezici sa najbržim rastom broja korisnika u 2018. godini su Go, TypeScript, Kotlin i Rust.

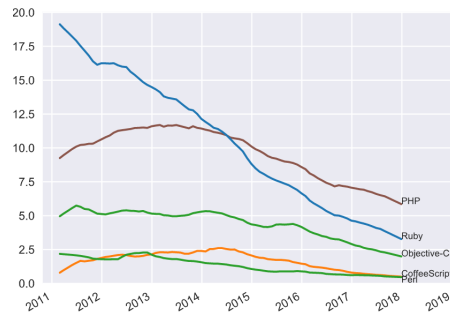


Kotlin se uglavnom koristi za razvoj Android aplikacija. Na grafu se može videti nagli porast korisnika kada je Google najavio punu podršku Kotlin-a na Android-u početkom 2017. godine.

Jedna zanimljiva stvar koja je zajednička za ove programske jezike je da ih sve sponzorishu velike kompanije: Google je pokrenuo Go, Microsoft TypeScript, JetBrains Kotlin i Mozilla Rust.

Stvaranje novog programskog jezika nije samo napraviti elegantan programski jezik za upotrebu. Treba razviti zajednicu i ekosistem iza jezika. Stvari kao što su podrška razvojnog okruženja, biblioteke i paketi za uobičajne zadatke, alati i dokumentacije pomažu u stvaranju jezika i privlačenju velikog broja korisnika.

3.3 Jezici koji su doživeli pad



Jezici kojima je opao broj korisnika u 2018. godini su Ruby, PHP, Objective-C, CoffeeScript i Perl. Ruby je zabeležio najveći pad u 2018. godini. Od 2. najpopularnijeg jezika u 2011. godini sa preko 18% korisnika do 11. najpopularnijeg u 2018. sa svega 3,2% korisnika.

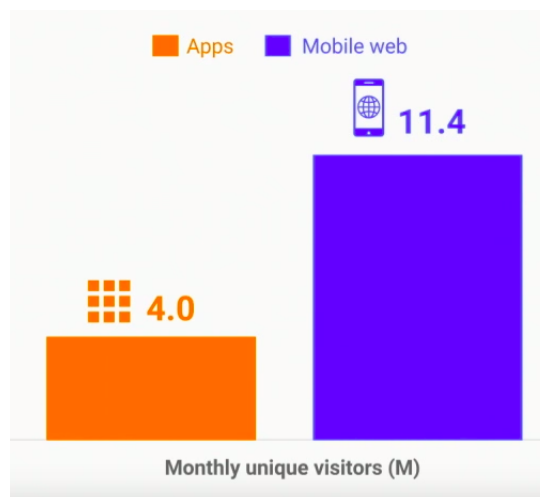
Pad korišćenja Objectiv-C odgovara porastu korišćenja Swift-a, dok je CoffeeScript zamenjen TypeScript-om. Dok pad korisnika Objectiv-C opada, čini se da je ukupan razvoj iOS aplikacija stabilan. Isto tako CoffeeScript je obezbedio put TypeScript-u tako što je navikao programere na ideju prevodenja JavaScript koda.

4 Trendovi i noviteti

U okviru ove sekcije osvrnućemo se na trendove, ali i na neke novitete koji su uvedeni u svetu programiranja u 2018. godini. Trendovi nam dosta govore o razvoju i popularnosti kako programskih jezika tako i koncepta koji se koriste prilikom programiranja. U protekloj godini su se u programiranju izdvojili određeni trendovi i pojavili novi ili ustalili neki koncepti koji su zaživeli u proteklih par godina. Neki od ovih trendova se ne odnose na konkretne programske jezike, ali svakako utiču na njihov razvoj i primenu.

4.1 Progressivne web aplikacije

Jedan od trendova koji se javio u Web programiranju jesu **progressivne web aplikacije** (PWA) koje predstavljaju web-sajtove koji funkcionišu kao da su u pitanju mobilne aplikacije. Dakle, ove web aplikacije imaju sve mogućnosti, informacije i funkcionalnosti kao i odgovarajuća mobilna aplikacija. Progressivne web aplikacije prvenstveno imaju značaj jer su vezane za web, najveći platformu na svetu, pa im je lakše i pristupiti što govore i istraživanja, pa na osnovu istraživanja Google-a mesečno 3 puta više posetioca imaju pwa od mobilnih aplikacija (što je i predstavljeno na sledećoj slici).



PWA su vremenom sve popularnije i među programerima što se može videti na osnovu pretraživanja koja su vezana za PWA. Grafikon na slici 9 predstavlja interesovanje u procentima za odgovarajući termin.



Slika 9: Interesovanje za progresivne web aplikacije

Neke prednosti progresivnih web aplikacija su:

- Pozdanost - odmah se učitavaju i to nezavisno od stanja interneta
- Brzina - brzi odgovori prilikom interakcije sa korisnikom
- Pogodnosti za korišćenje - mogu se instalirati i biti na početnim ekranima korisnika bez potrebe za skidanjem sa Play Store-a ili App Store-a. Ovu funkcionalnost omogućava Web App Manifest.
- Raspoloživost - mogu se pokrenuti na mobilnim uređajima, stonim računarima i tabletima

Progresivne web aplikacije su se dobro pokazale i u praksi što dokazuje i to da je nakon ponovnog pokretanja BMW.com kao progresivna web aplikacija primećeno 3-4 puta brže učitavanje, 30% više klikova i 26% više mobilnih korisnika. Isto tako, kada je Trivago predstavio svoju novu progresivnu web aplikaciju, kompanija je zabeležila 50% povećanje mobilnih sesija zajedno sa 150% povećanjem aktivnosti korisnika koji su dodali Trivago na početnu stranicu.

U svom izveštaju za 2017. godinu Gartner je predvideo da će PWA biti najveći trend razvoja softvera u narednim godinama što se obistinilo u 2018. godini.

4.2 Blockchain tehnologija

Bitcoin i druge blockchain valute mogu imati svoje uspone i padove na tržištu, ali blockchain tehnologija dobija na snazi, posebno u finansijskom sektoru. Neki od slučajeva upotrebe koji ukazuju na svetlu budućnost ove tehnologije su:

- Distribucija resursa
- Upravljanje identitetom
- Pametni ugovori
- Podaci o zdravstvenoj zaštiti

Među velikim kompanijama koje su ušle u blockchain su Amazon, Microsoft, Facebook, IBM. Ovo takođe govori da možemo da očekujemo rast ove tehnologije i u predstojećim godinama.

4.3 Low-code razvoj

Sve se više teži da se omogući izgradnja aplikacija i onima koji nisu tehnički podkovani. Ideja je da zaposleni u kompanijama mogu da izrađuju poslovne aplikacije koje odgovaraju specifičnim potrebama preduzeća, koristeći vizualno programiranje, drag-and-drop module i automatsko generisanje koda. Ovakav razvoj aplikacije je poznat pod nazivom Low-code razvoj (LCD).

Forrester, američka kompanija za istraživanje tržišta koja pruža savete o postojećem i potencijalnom uticaju tehnologije svojim klijentima i javnosti, procenjuje da će u 2019. godini platforme za low-code razvoj stvoriti prihod od 10 milijardi dolara, što je skoro duplo više u odnosu na 2018. godinu. Ovakav koncept programiranja otvara vrata netehničkim licima i dosta ubrzava izradu, pa se samim tim smanjuje i vreme koje je potrebno da se izradi aplikacija koja je spremna za produkciju i punu upotrebu.

Naravno LCD nije rešenje koje je poželjno i pogodno u svakoj situaciji. Za neke složenije zadatke, prilagođeni softver je mnogo bolji izbor.

Sa druge strane, radni proces sa dobro definisanim ciljevima I jasnom strukturom je idealan za implementaciju aplikacija izgrađenih koristeći platforme za LCD. Grafikon na slici 10 predstavlja interesovanje u procentima za odgovarajući termin.



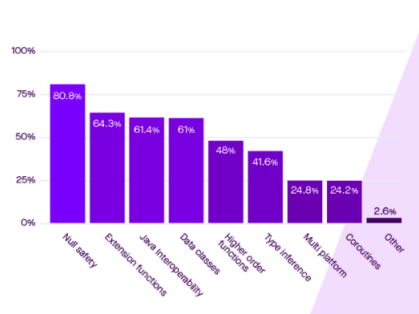
Slika 10: Interesovanje za low-code razvoj

4.4 Noviteti u programiranju Android aplikacija

Što se tiče noviteta u programiranju Android aplikacija treba posebno spomenuti programski jezik Kotlin, koji je 2017. i 2018. godine doživeo veliki uspeh, ali i uveo mnoge novitete koji olakšavaju život programerima, ali takođe pružaju pregršt pogodnih funkcija. Ova jezik je dobio ogromnu pažnju od Android programera i u narednim godinama će sigurno uživati veliku popularnost. Ima dosta sličnosti sa Javom, a jedan od najvećih noviteta koji olakšava programiranje je to da je Kotlin, za razliku od Jave bezbedan što se tiče null vrednosti. Naime, Java omogućava dodeljivanje null vrednosti bilo kojoj promenljivoj, ali prilikom korišćenja reference objekta koji ima null vrednost, onda dolazi do pojave koja najviše frustrira Java programere, a to je NullPointerExceptions. Sa druge strane, u Kotlin-u, svi tipovi su podrazumevano non-nullable tj ne mogu imati null vrednost. Ako pokušamo da dodelimo ili vratimo null u naš Kotlin kod, greška će biti prijavljena za vreme kompajliranja.

Još jedan novitet koji uvodi Kotlin jesu "coroutine". Naime, kada imamo neku dugotrajnu operaciju, nit se blokira dok se ta operacija ne završi, a čim se glavna nit blokira korisnički interfejs će se zamrznuti I aplikacija će ostati neaktivna dok se operacija ne završi. U Javi, rešenje je bilo da se napravi pozadinska nit u kojoj se može izvesti ova dugotrajna operacija, ali upravljanje višestrukim nitima može dovesti do složenog koda koji je sklon greškama, a i kreiranje nove niti je skupa operacija. Ovde na scenu stupaju "coroutine" Kotlina koje obavljaju dugotrajne i intezivne zadatke obustavljanjem izvršavanja u određenoj tački bez blokiranja niti, a zatim nastavljanje ove funkcije u sledećoj tački, moguće i na drugoj niti. Ovo nam omogućava da kreiramo asinhroni kod bez blokiranja koji izgleda sinhrono i dosta je jasniji, koncizniji i čitljiviji. "Coroutine" bez steka, tako da upotrebljavaju manje memorije u odnosu na niti i otvaraju vrata dodatnim stilovima asinhronog programiranja bez blokiranja, kao što je `async/await`.

Kotlin uvodi još nekoliko noviteta kao što su pametna konverzija tipova (Smart Cast), dodavanje novih metoda na već postojeće tipove ("Extension functions") itd. Na slici možete videti koje su karakteristike, koje je Kotlin uveo, programeri ocenili kao najbolje I najznačajnije.

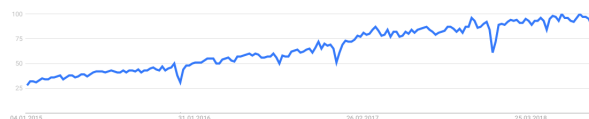


Slika 11: Ocena karakteristika programskog jezika Kotlin

4.5 Front end JavaScript okviri

Konačno, bez spominjanja React-a i Angular-a priča o trendovima u programiranju u poslednje vreme je nepotpuna. React i Angular su sve više prihvaćeni od strane dizajnera za dizajn i izradu prototipova korisničkog interfejsa. Oni nude arhitekturu zasnovanu na komponentama i modulima, što dosta utiče na strukturu samih projekata. Njihovom pojavom dizajn sve više postaje baziran na komponentama a ne orijentisan na ekran. Za Angular je 2018. godina bila od posebnog značaja jer su u 2018. godini izdali novu verziju (Angular 7).

Ono što je veoma interesantno je i to što iako se i React i Angular koriste za praktično istu stvar, popularnost i jedanog i drugog okvira su i dalje u konstantnom porastu, što se može videti i na sledećim grafikonima.



Slika 12: Interesovanje za Angular u prethodnom periodu



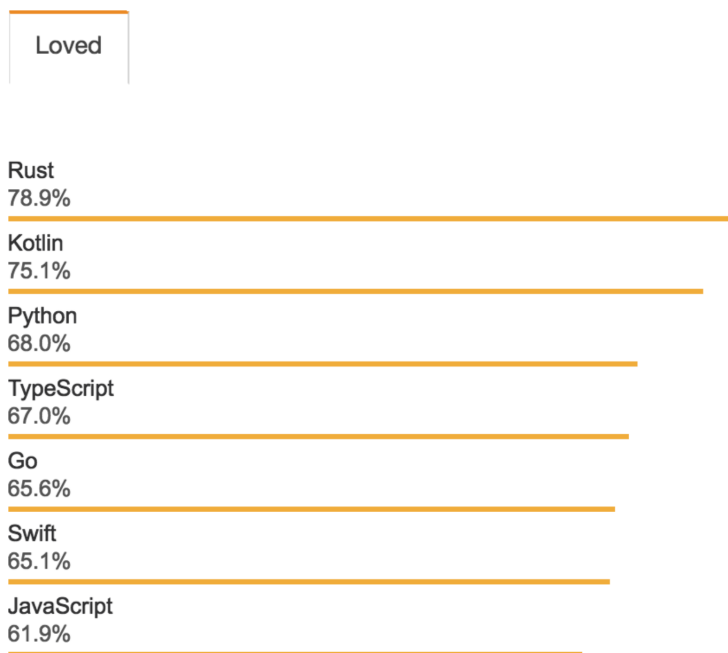
Slika 13: Interesovanje za React u prethodnom periodu

5 Dva novija jezika koja smo izdvojili

Ovu sekciju smo izdvojili da bismo vas bolje upoznali sa tri jezika vrednih pomena. Trenutno se ne nalaze na spisku najtraženijih jezika, nisu najviše plaćeni, ali su noviji i sa tendencijom da njihova upotreba drastično poraste.

GO. Obaveštenje o jeziku je prvi put izneto 2009. godine a prva verzija je nastala 2012. godine od strane Google-a. Reč je o jeziku na sistemskom nivou. Koristi se za velike distribuirane sisteme i mrežne servere. Planiran je da zameni C++ i Javu u svrhe Google-a. Go bi pomogao u rešavanju problema kao što su sporo kompajliranje, kako uklopiti programere koji koriste različite programske jezike, citljivij kod, bolja dokumentacija, problemi usled različitih verzija, pokretanje programa na različitim jezicima...

Prema analizama na sajtu stackoverflow, Go se pokazao kao jedan od najvoljenijih jezika medju programerima. Statistika pokazuje da 65.6% programera koji su koristili Go bi radije ostali da ga koriste nego da promene tehnologiju. Slika 14.



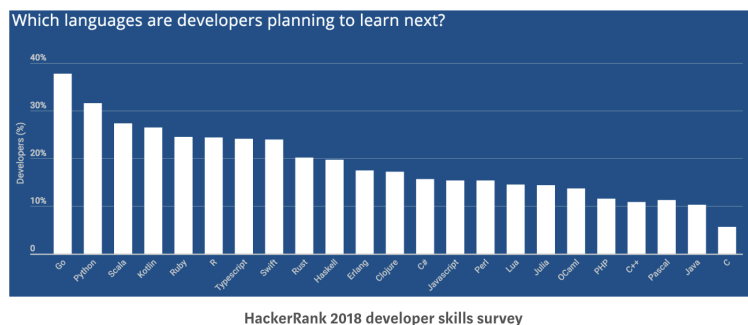
Slika 14: Najvoljeniji jezici po Stack Overflow-u

Ukoliko se osvrnemo na sajt GitHub koji smo već pominjali kao najveću bazu programa otvorenog koda, možemo zapaziti da se 2016. godine na sajtu nalazilo oko 188.000 otvorenih "pull request", da bi naredne 2017. godine taj broj iznosio oko 285.000 i time postao deveti najpopularniji jezik na GitHub.

Na sajtu hackerrank, takodje veoma popularan medju programerima, možemo da primetimo koliko programera se odlučilo za GO kao narednu opciju za učenje. Slika 15

Zašto je sistemskog nivoa jezik tako popularan?

Jednostavnost je verovatno glavni razlog za brzo probijanje GO. Ako posmatramo razvoj drugih jezika, konstantno se ubacuju nove upotrebe



Slika 15: Prema sajtu hackerrank, sledeći jezik po izboru programera za učenje

i načini korišćenja koje uvek zvuče kao da je dobro nešto dodato. Veliki problem je u tome što jezici postaju sve sličniji u korišćenju, upotrebi i osećaju rada u njima. GO je upravo jedinstven, takav kakav je i ne pokušava da bude nešto drugo. Glavna ideja osnivača je bila da jezik bude čitljiv i jednostavan. Vodeći se tom idejom, pretpostavljamo da neće biti bitnih promena u GO što se tiče funkcionalnosti i načina korišćenja dug vremenski period. Jedini preporučeni napredak koji će biti dodat je "generics", ali nije hitno. Postoji mogućnost nastajanja GO2 ali ne u narednih nekoliko godina.

Šta je jedinstveno za GO?

GO se zasniva na brzom kompajliranju. Jezik u potpunosti apstrahuje upravljanje šmećima i ručno upravlja memorijom što mnogo smanjuje vreme potrebno da se program kompajlira. Upravo to oduzima mnogo vremena kod programa pisanih u C ili C++. Go nema klasične klase, sve se svodi na strukture. Statički je tipiziran jezik pa se zato većina grešaka javljaju u vremenu kompajliranja a ne u radu. Go je takodje i "type-safe" što znači da ne možemo da se igramo sa tipovima na način na koji bismo pokvarili program. Go je i "memory-safe" što znači da ne moramo ručno da vodimo računa o prekoračenjima, baferima, pokazivačima... U mnogim drugim jezicima bismo morali da biramo throw/return exception/error, ali ovde možemo i da vratimo "error" i vrednost istovremeno. Ovo je bitno jer GO ne poseduje izuzetke. Možda ovo zvuči kao ograničavajući faktor, ali može biti veoma oslobadjajuće i moćno u pravim situacijama.

```

type MyError struct {
    What string
}

func (e MyError) Error() string {
    return fmt.Sprintf("%v", e.What)
}

func someError() error {
    return MyError{"Some error occurred"}
}

func main() {
    if err := someError(); err != nil {
        fmt.Println(err)
    }
}

```


Funkcije mogu biti prosledjene drugim funkcijama. Funkcija može biti i vraćena iz funkcije, a može se i deklarirati kao promenljiva. Takođe, funkcija se može i definisati unutar neke druge funkcije.

```
package main

import (
    "fmt"
)

func counter() func() int {
    i := 0
    return func() int {
        i++
        return i
    }
}

func doubleUp(current int) int {
    currentCount := current
    return currentCount * 2
}

func main() {
    someCounter := counter()

    // prints "1"
    fmt.Println(someCounter())

    // prints "4"
    // because someCounter() increases i to 2
    // then doubleUp() doubles what was returned
    fmt.Println(doubleUp(someCounter()))

    // prints "3"
    // because the previous doubleUp() didn't affect i
    // only the value returned from someCounter
    fmt.Println(someCounter())
}
```

GO poseduje ključnu reč `defer`. To znači da možemo da sačekamo na izvršavanje unutrašnje funkcije dok spoljašnja ne vrati rezultat. Ta funkcija će se izvršiti momentalno ako može, ali ne pre nego što se spoljašnja ne završi. Pozivi `defer` funkcija se izvode u redosledu "last in first out".

```
package main

import (
    "fmt"
)

// Will print "0"
// "i" is evaluated at defer
func a() {
    i := 0
    defer fmt.Println(i)
    i++
    return
}

// Will print "3210"
// Executes in LIFO
func b() {
```

```

    for i := 0; i < 4; i++ {
        defer fmt.Println(i)
    }
}

// Will return 2
// "i" is incremented after return
func c() (i int) {
    defer func() { i++ }()
    return 1
}

func main() {

    a()
    b()
    fmt.Println(c())
}

/*
Output:
0
3
2
1
0
2
*/

```

Jupyter Notebook - Prvi put smo se sreli sa Jupyter Notebook pre 5 godina tada poznatim kao IPython notebooks, program za organizaciju i analizu podataka. Od tada, Jupyter ima neverovatnu statistiku. Koristeći Google BigQuery na javnim podacima možemo pronaći koliko puta je Jupyter Python paket instaliran na PyPi (centralizovan repoiztorijum za Python paketne distribucije).

```

#https://langui.sh/2016/12/09/data-driven-decisions/
SELECT
    STRFTIME_UTC_USEC(timestamp, "%Y-%m") AS yyyyymm,
    COUNT(*) as download_count
FROM
    TABLEDATE_RANGE(
        [the-psf:pypi.downloads],
        DATEADD(CURRENT_TIMESTAMP(), -1, "year"),
        CURRENT_TIMESTAMP()
    )
WHERE file.project="jupyter"
GROUP BY yyyyymm
ORDER BY yyyyymm DESC
LIMIT 12

```

Rezultat na slici.

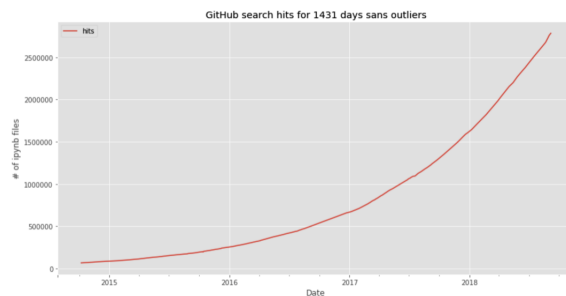
Row	yyyymm	download_count	
1	2018-09	262745	
2	2018-08	648689	
3	2018-07	189518	
4	2018-06	172282	
5	2018-05	219907	
6	2018-04	230715	
7	2018-03	248621	
8	2018-02	254697	

Google BigQuery baza podataka takodje uključuje i podatke sa Stack Overflow-a, pa bi naredni query mogao da nam pokaže potražnju u pitanjima na Stack Overflow-u za Jupyter.

```
SELECT tags, COUNT(*) c, year
FROM (
  SELECT SPLIT(tags, '|') tags, YEAR(creation_date) year
  FROM [bigquery-public-data:stackoverflow.
  posts_questions] a
  WHERE YEAR(creation_date) >= 2014 AND tags LIKE '%
  jupyter-notebook%'
)
WHERE tags='jupyter-notebook'
GROUP BY year, tags
ORDER BY year DESC
```

Row	tags	c	year	
1	jupyter-notebook	2247	2018	
2	jupyter-notebook	2064	2017	
3	jupyter-notebook	1178	2016	
4	jupyter-notebook	51	2015	
5	jupyter-notebook	15	2014	

Ako ispratimo kretanje Jupyter-a na GitHub-u, sajt nam obezbedjuje grafik gde vidimo kretanje količine Jupyter-ovih fajlova, tj fajlova sa ekstenzijom .ipynb.



Nagli razvoj Jupyter doživljava tek u poslednjih nekoliko godina. Naziv je dobio jer su osnivači pronašli inspiraciju u Julia (Ju), Pythonu (Py) i R. Za obradu podataka Jupyter je toliko napredovao da je postao standard u toj oblasti, po rečima Lorena Barba, mehanički inženjer na George Washington University.

Jupyter ima dve komponente. Unos korisnika programerski kod ili tekst u ćelije na fron-end strani. Pretraživač prosledjuje kod na back-end (kernel) koji pokreće kod i vraća rezultat. Prema osnivaču Perezu, više od 100 Jupyter kernela je napravljeno, koji podržavaju veliku količinu različitih jezika. Notebook može takodje da se pokreće i na cloud-u. Najnovija varijanta Jupytera jeste JupyterLab, čija beta verzija je nastala u januaru 2018. godine i dostupna je ili kao samostalan paket ili kao deo besplatnog Anaconda okruženja.

Literatura

- [1] <https://medium.com/qash/golang-the-next-language-to-learn-for-developers-2c7c32b8a095>
- [2] <https://hackernoon.com/top-3-most-popular-programming-languages-in-2018-and-their-annual-salaries-51b4a7354e06>
- [3] <https://medium.freecodecamp.org/best-programming-languages-to-learn-in-2018-ultimate-guide-bfc93e615b35>
- [4] <https://www.codingdojo.com/blog/7-most-in-demand-programming-languages-of-2018>
- [5] <https://dotnet.microsoft.com/languages>
- [6] <https://developer.microsoft.com/en-us/web/platforms-tech/>
- [7] <https://blog.ouseful.info/2018/09/10/the-growing-popularity-of-jupyter-notebooks/>
- [8] <https://www.nature.com/articles/d41586-018-07196-1>
- [9] <https://www.technotification.com/2018/12/programming-languages-google.html>
- [10] <https://stackoverflow.blog/2017/09/06/incredible-growth-python/>
- [11] <https://stackoverflow.blog/2017/10/10/impressive-growth-r/>
- [12] <https://www.benfrederickson.com/ranking-programming-languages-by-github-users/>
- [13] <https://trends.google.com>
- [14] <https://dzone.com/articles/the-rise-of-kotlin-the-programming-language-for-th>
- [15] <https://www.cmarix.com/programming-in-2019-7-key-trends-to-watch-out/>
- [16] <https://polcode.com/blog/software-development-trends-for-2019/>