

Najkorišćeniji razvojni alati

Seminarski rad u okviru kursa

Metodologija stručnog i naučnog rada

Matematički fakultet

Staša Đorđević, Lazar Savić, Đurđa Milošević, Bogdan Tomić
mi251007@alas.matf.bg.ac.rs, mi251004@alas.matf.bg.ac.rs,
mi251008@alas.matf.bg.ac.rs, mi251040@alas.matf.bg.ac.rs

25. decembar 2025.

Sažetak

Istraživanjem je analizirano korišćenje razvojnih alata u programiranju sa ciljem identifikovanja faktora koji utiču na njihov izbor i promenu. Podaci su prikupljeni anonimnom anketom. Rezultati pružaju uvid u aktuelne trendove i razloge preferiranja pojedinih razvojnih okruženja i sistema za verzionisanje.

Sadržaj

1 Uvod	2
2 O razvojnim alatima	2
2.1 Integrisana razvojna okruženja (IDE)	2
2.2 Tekstualni editori koda	3
2.3 Sistemi za verzionisanje	3
3 Metodologija	3
3.1 Uzorkovanje	3
3.2 Struktura uzorka	3
3.3 Struktura ankete	4
4 Rezultati	4
4.1 Kriterijumi pri izboru alata	4
4.2 Opšte preferencije Git platformi	6
4.3 Uticaj radnog iskustva na promenu razvojnog alata	6
4.4 Uticaj navike na promenu razvojnog alata	7
4.5 Uporedna analiza prioriteta: Studenti nasuprot zaposlenih .	8
4.6 Analiza otvorenih odgovora ispitanika	8
5 Diskusija	9
5.1 Poređenje dobijenih rezultata sa literaturom	9
5.2 Faktori koji utiču na izbor alata	10
5.3 Ograničenja istraživanja	10
6 Zaključak	11
Literatura	11

1 Uvod

Razvojni alati predstavljaju jedan od najznačajnijih faktora koji utiču na produktivnost i efikasnost programera, i kao takvi, čine neizostavni deo njihovog svakodnevnog rada. Imajući u vidu veliki broj dostupnih razvojnih alata, koji se međusobno razlikuju po funkcionalnostima, performansama, mogućnostima prilagođavanja i korisničkom iskustvu, izbor najkorisnijeg alata nije jednoznačan. On često zavisi od ličnih preferencija korisnika, prethodnog iskustva, kao i specifičnih zahteva projekta ili okruženja u kojem se alat koristi.

Predmet ovog rada jeste analiza upotrebe razvojnih alata, sa fokusom na integrisana razvojna okruženja (IDE), tekstualne editore koda i sisteme za verzionisanje, kao tri osnovne kategorije alata koje se najčešće koriste u procesu razvoja softvera. Cilj rada jeste da se ispita koje konkretnе alate iz navedenih kategorija studenti i iskusniji programeri najčešće koriste, koje faktore smatraju najvažnijim prilikom njihovog izbora, kao i šta ih motiviše da pređu na novi alat ili ostanu pri postojećem. Pored toga, poseban akcenat stavljen je na ulogu navike u korišćenju određenog alata, kao i na zadovoljstvo korisnika alatima koje trenutno koriste, u cilju identifikovanja potencijalnih unapredena samih alata.

Značaj ovog istraživanja ogleda se u pružanju sistematičnog uvida u način na koji korisnici percipiraju i biraju razvojne alate, kao i u boljem razumevanju faktora koji utiču na donošenje odluka u procesu izbora alata u realnom akademskom i profesionalnom kontekstu.

2 O razvojnim alatima

Razvoj softvera se oslanja na upotrebu različitih alata koji programerima omogućavaju efikasnije pisanje, organizaciju, testiranje i održavanje programskog koda. Razvojni alati imaju značajan uticaj na produktivnost i način rada programera, pri čemu empirijske studije ukazuju da, iako savremena integrisana razvojna okruženja unapređuju pojedine aspekte rada, njihova rastuća kompleksnost može dovesti do povećanih troškova učenja i debagovanja [3]. U ovom radu razmatraju se tri osnovne kategorije razvojnih alata: integrisana razvojna okruženja (IDE), tekstualni editori koda i sistemi za verzionisanje.

2.1 Integrisana razvojna okruženja (IDE)

Integrisana razvojna okruženja (IDE – *Integrated Development Environment*) predstavljaju softverske alate koji objedinjuju više funkcionalnosti potrebnih za razvoj softvera u jedinstveno radno okruženje [7]. Tipične komponente IDE-a uključuju editor koda, kompajler ili interpreter, alate za dinamičku analizu koda, kao i dodatne funkcionalnosti za upravljanje projektima i testiranje. Primeri integrisanih razvojnih okruženja koji se često koriste u praksi uključuju **Visual Studio**, **IntelliJ IDEA**, **PyCharm** i **Android Studio**.

Glavna prednost integrisanih razvojnih okruženja ogleda se u visokom stepenu automatizacije i podrške programeru. Funkcionalnosti kao što su automatsko dopunjavanje koda, staticka analiza, upozorenja na potencijalne greške i integrisana dokumentacija mogu značajno ubrzati proces razvoja i smanjiti broj grešaka u kodu. Sa druge strane, integrisana razvojna okruženja mogu zahtevati veće hardverske resurse i biti složenija za prilagođavanje individualnim potrebama korisnika.

2.2 Tekstualni editori koda

Tekstualni editori koda predstavljaju alate namenjene pisanju i uređivanju izvornog koda, sa fokusom na jednostavnost i fleksibilnost [1]. Za razliku od integrisanih razvojnih okruženja, tekstualni editori obično nude osnovne funkcionalnosti, kao što su isticanje sintakse, jednostavna navigacija kroz fajlove i minimalna podrška za analizu koda.

Jedna od ključnih prednosti tekstualnih editora jeste njihova brzina rada i mala potrošnja sistemskih resursa. Mnogi editori omogućavaju proširivanje funkcionalnosti putem dodataka ili ekstenzija, čime se mogu prilagoditi različitim programskim jezicima i stilovima rada. Primeri često korišćenih tekstualnih editora koda su **Sublime Text**, **Kate** i **Vim**.

2.3 Sistemi za verzionisanje

Sistem za verzionisanje je sistem koji je zadužen za čuvanje i kontrolisanje izmena programskog koda, odnosno za upravljanje različitim verzijama softvera koji se razvija [6]. Njihova osnovna uloga jeste čuvanje istorije izmena, omogućavanje povratka na prethodne verzije koda i olakšavanje saradnje više korisnika na istom projektu.

Dele se na centralizovane, koji se oslanjaju na centralni repozitorijum sa kompletnom istorijom izmena, i distribuirane, kod kojih svaki korisnik poseduje lokalnu kopiju repozitorijuma. Najpoznatiji primer distribuiranog sistema za verzionisanje je **Git**.

3 Metodologija

Istraživanje je sprovedeno korišćenjem anonimne onlajn ankete. Metodološki pristup je dominantno kvantitativan, budući da su podaci prikupljeni putem strukturiranih pitanja i analizirani primenom deskriptivne statistike, uz ograničene kvalitativne uvide dobijene analizom odgovora otvorenog tipa.

3.1 Uzorkovanje

U istraživanju je učestvovalo ukupno 43 ispitanika koji se bave programiranjem. Uzorak je obuhvatio studente tehničkih i srodnih fakulteta, kao i zaposlene profesionalce u IT sektoru. Uzorkovanje je sprovedeno metodom pogodnog (neprobabilističkog) uzorka, s obzirom na to da je anketa distribuirana putem onlajn kanala i da su ispitanici dobrovoljno učestvovali u istraživanju, uz garantovanu anonimnost odgovora.

3.2 Struktura uzorka

Struktura uzorka analizirana je kroz demografske i profesionalne karakteristike ispitanika, pri čemu su prikupljeni podaci o dužini iskustva u programiranju, pohađanim fakultetima, kao i polu. Među ispitanicima, 27 je bilo muškog, a 16 ženskog pola. Ubedljivo najviše ispitanika je sa Matematičkog fakulteta 79,1%, dok je zastupljenost ostalih fakulteta mala.

Kada je u pitanju dužina iskustva u programiranju (uključujući školu, fakultet, lične projekte i profesionalni rad), najveći deo uzorka čine ispitanici sa srednjim nivoom iskustva: čak 51,2% ispitanika ima između 3 i 5 godina iskustva, dok 32,6% ima između 6 i 9 godina iskustva. Manji,

ali značajan deo uzorka čine iskusniji programeri sa 10 do 15 godina iskustva (14%). Ostale kategorije (manje od jedne godine i više od 15 godina iskustva) nisu statistički značajne u uzorku.

3.3 Struktura ankete

Anketa je sadržala 16 pitanja, od kojih je 12 bilo zatvorenog tipa, dok su četiri pitanja omogućavala slobodne odgovore. Pitanja su organizovana po celinama:

- **Demografska i profesionalna pitanja** - dužina iskustva u programiranju, pol, pohađani fakultet, trenutna pozicija u IT sektoru. Pitanja su bila zatvorenog tipa sa ponuđenim odgovorima.
- **Pitanja o korišćenju razvojnih alata** - najčešće korišćeni IDE i tekstualni editori, sistemi za verzionisanje, preferencije između GitHub i GitLab platforme, glavni razvojni alat, razlozi izbora i problemi sa alatima. Većina pitanja bila je zatvorenog tipa, dok su argumentacija izbora i opis problema bili otvorenog tipa radi dodatnih komentara.
- **Faktori izbora razvojnog alata** - pitanja o važnosti karakteristika alata, uticaju dugogodišnjih navika, ranijim promenama glavnog alata, razlozima za prelazak na novi alat i faktorima koji bi motivisali tu promenu. Za većinu ovih pitanja korišćena je *Likert skala* od 1 do 5.

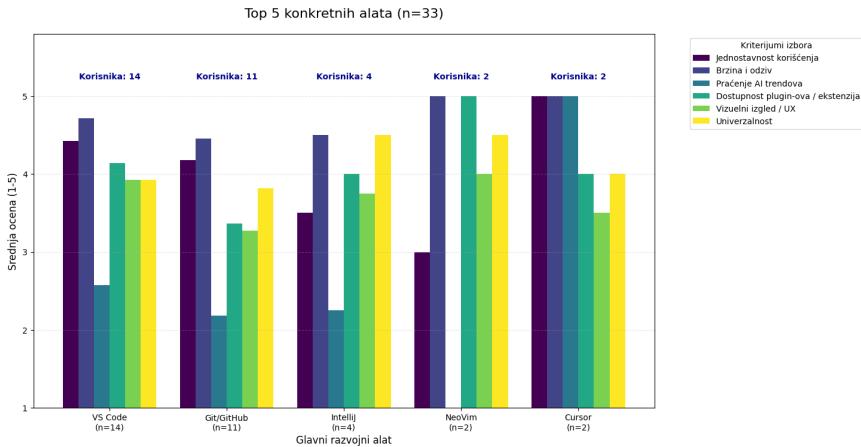
Svi odgovori su bili anonimni, a učestvovanje u anketi dobrovoljno.

4 Rezultati

U ovom poglavlju analizirani su podaci prikupljeni istraživanjem, uz poseban osvrт na korelacije između različitih parametara. Budući da uzorak obuhvata učesnike različitih nivoa stručnosti i obrazovnih profila, stanje na tržištu je sagledano iz više perspektiva. Vizuelni prikazi generisani su direktno iz prikupljenih odgovora.

4.1 Kriterijumi pri izboru alata

Fokus ovog dela rada je na kriterijumima koje programeri smatraju presudnim prilikom odabira radnog okruženja. Ispitivani su sledeći faktori: jednostavnost korišćenja, brzina i odziv okruženja, praćenje i integracija AI trendova, dostupnost dodataka i ekstenzija, vizuelni izgled i UX dizajn, kao i univerzalnost alata. Na slici 1 prikazani su rezultati za pet najzastupljenijih alata.



Slika 1: Važnost faktora prilikom izbora razvojnog alata (Top 5)

Za preciznije rangiranje uveden je pokazatelj reprezentativnosti alata P_A , koji ponderiše prosečnu ocenu faktora sa udelom korisnika tog alata u ukupnom uzorku. Formula za izračunavanje glasi:

$$P_A = \left(\frac{f_{jed} + f_{brz} + f_{AI} + f_{plg} + f_{viz} + f_{uni}}{6} \right) \cdot \frac{n_A}{N_{top5}} \quad (1)$$

Značenje parametara korišćenih u formuli detaljno je opisano u tabeli 1.

Tabela 1: Opis parametara formule za reprezentativnost alata

Parametar	Opis faktora / Vrednost
f_{jed}	Jednostavnost korišćenja
f_{brz}	Brzina i odziv okruženja
f_{AI}	Praćenje i integracija AI trendova
f_{plg}	Dostupnost dodataka i ekstenzija
f_{viz}	Vizuelni izgled i UX dizajn
f_{uni}	Univerzalnost alata
n_A	Broj korisnika koji koriste alat A
N_{top5}	Broj korisnika nekih od top 5 alata ($N = 33$)

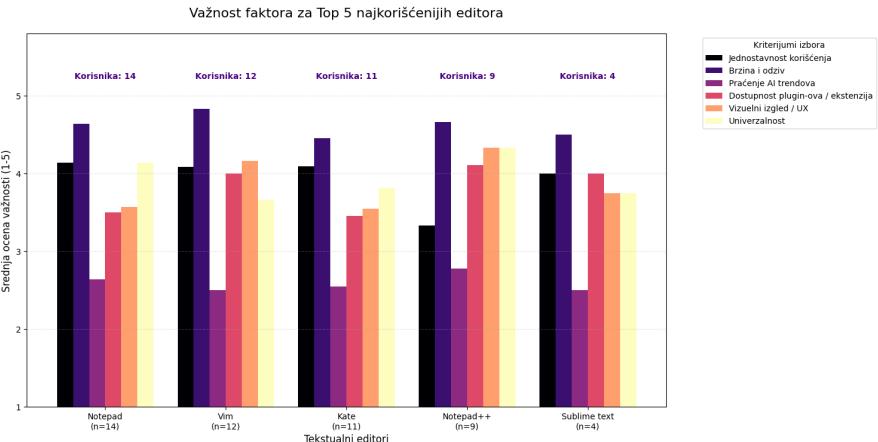
Na osnovu prikupljenih podataka, u tabeli 2 sumirani su rezultati prosečnih ocena (\bar{P}_A) i ključne karakteristike za svaki od analiziranih alata.

Tabela 2: Analiza vodećih razvojnih alata na osnovu pojedinačnih faktora važnosti

Alat	n	f_{jed}	f_{brz}	f_{AI}	f_{plg}	f_{viz}	f_{uni}	\bar{P}_A
VS Code	14	4.4	4.7	2.6	4.1	3.9	3.9	1.67
Git/GitHub	11	4.2	4.5	2.2	3.4	3.3	3.8	1.19
IntelliJ	4	3.5	4.5	2.3	4.0	3.8	4.5	0.46
Cursor	2	5.0	5.0	5.0	4.0	5.0	5.0	0.29
NeoVim	2	3.0	5.0	2.5	5.0	4.0	4.5	0.24
Prosečna ocena faktora		4.02	4.74	2.92	4.10	4.00	4.34	—

Iz prikazanih rezultata uočava se da korisnici najviše vrednuju brzinu i odziv razvojnog okruženja, dok faktor integracije AI tehnologija ostvaruje najnižu prosečnu ocenu.

U okviru ove analize odrđena je i analiza koja se vezuje isključivo za tekstualne editore koda. Na slici 2 prikazano je kako korisnici pet najzastupljenijih editora ocenjuju ključne faktore prilikom izbora svog primarnog alata. Na osnovu prikupljenih odgovora, analizirane su prosečne vrednosti šest ključnih kriterijuma, pomenutih na početku ovog odeljka.



Slika 2: Važnost faktora izbora najčešće korišćenih tekstualnih editora

Ova analiza potvrđuje da korisnici tekstualnih editora absolutni prioritet daju performansama i brzinu rada. Iako su alati poput *Vim-a* i *NotePad++* ocenjeni stabilno u svim kategorijama, zajednička nit svim ispitanicima je niska važnost AI trendova u kontekstu čistih tekstualnih editora, gde se primat ostavlja stabilnosti i efikasnosti.

4.2 Opšte preferencije Git platformi

U okviru istraživanja, jedan od ciljeva bio je i utvrđivanje popularnosti vodećih platformi za upravljanje verzijama koda: *GitHub-a* i *GitLab-a*. Analiza je sprovedena na celokupnom uzorku ispitanika.

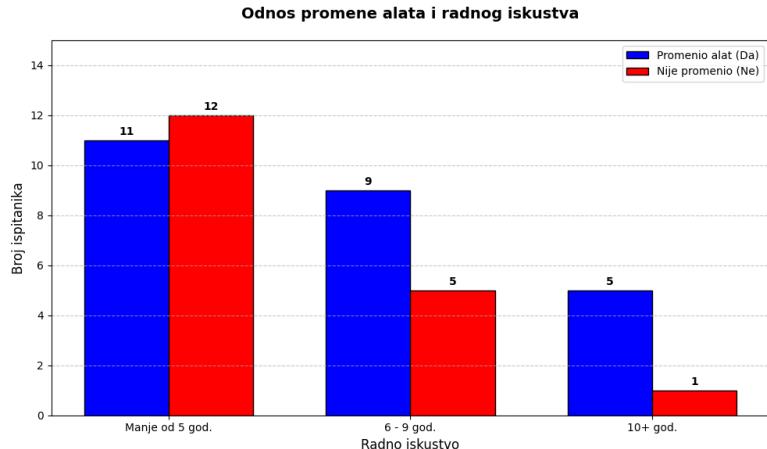
- Apsolutna dominacija GitHub platforme:** Najveći deo uzorka, tačnije 88,4% (ukupno 38 ispitanika), opredelio se za *GitHub*.
- Zastupljenost GitLab-a:** Znatno manji ideoe ispitanika, odnosno 11,6% (ukupno 5 ispitanika), koristi *GitLab* kao primarnu platformu.

Odnos od skoro 9:1 u korist GitHub-a ukazuje na to da korisnici u velikoj većini biraju platforme koje omogućavaju najlakšu kolaboraciju i najveću bazu zajednice.

4.3 Uticaj radnog iskustva na promenu razvojnog alata

U ovom delu istraživanja analizira se korelacija između dužine radnog staža ispitanika i njihove tendencije ka prelasku sa jednog na drugi primarni razvojni alat. Rezultati prikazani na slici 3 pružaju uvid u to

koliko su programeri u različitim fazama karijere skloni usvajanju novih tehnologija.



Slika 3: Uticaj radnog iskustva na promenu razvojnog alata

Kako bi se jasnije uočili trendovi, podaci su grupisani u tri ključne kategorije, a rezultati ukazuju na sledeće:

- **Manje od 5 godina iskustva:** Ova objedinjena grupa (koja obuhva početnike i programere sa 3–5 godina staža) predstavlja najbrojniji deo uzorka. Primećuje se umerena podeljenost u navikama: od ukupno 23 ispitanika, 11 je promenilo alat, dok je 12 ostalo pri prvobitnom izboru.
- **Srednji nivo (6–9 godina iskustva):** Kod ovih ispitanika uočen je visok stepen prilagodavanja trendovima. Većina programera u ovoj grupi (9 od ukupno 14) odlučila se za promenu primarnog alata.
- **Iskusni kadar (10+ godina iskustva):** Grupa sa najdužim stažom pokazuje najveću proaktivnost. Od 6 ispitanika, čak 5 je prešlo na novo okruženje.

Ukupni podaci sugerisu da akumulacija radnog staža (posebno preko 6 godina) korelira sa većom verovatnoćom promene alata. Razlog za ovakav trend verovatno leži u kompleksnosti projekata, kao i u integraciji novih tehnologija poput veštačke inteligencije u modernim razvojnim okruženjima.

4.4 Uticaj navike na promenu razvojnog alata

Jedan od ciljeva istraživanja bio je ispitivanje uloge navike u procesu donošenja odluke o promeni glavnog razvojnog alata. Polazna pretpostavka bila je da će ispitanici kod kojih navika ima manji značaj biti skloniji promeni alata. Analizirana je veza između ocene uticaja navike na izbor razvojnog alata i odgovora ispitanika na pitanje da li su u prethodnom periodu menjali svoj primarni alat.

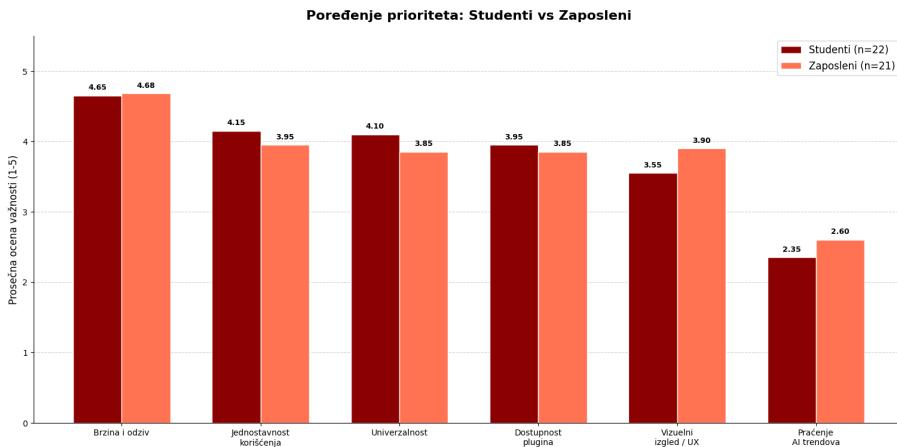
Analiza prosečnih ocena pokazuje da ispitanici koji su menjali glavni razvojni alat uticaj navike ocenjuju prosečnom ocenom od približno 4,1, dok ispitanici koji nisu menjali alat naviku ocenjuju prosečnom ocenom

od oko 4.0. Iako postoji razlika u prosečnim vrednostima između grupa, ona nije izražena, a posebno je uočljivo da ispitanici koji su menjali alat takođe navici pridaju veliki značaj. Na osnovu toga može se zaključiti da početna pretpostavka istraživanja u ovom slučaju nije potvrđena.

Dobijeni rezultati sugerisu da prelazak sa jednog razvojnog alata na drugi ne mora nužno biti povezan sa slabijim formiranim navikama, već pre sa spremnošću programera da postojeće navike prevaziđu u situacijama kada novi alat nudi jasne funkcionalne prednosti ili bolje odgovara zahtevima konkretnog projekta.

4.5 Uporedna analiza prioriteta: Studenti nasuprot zaposlenih

Ova podsekcija razmatra razlike u prioritetima između studenata ($n = 22$) i zaposlenih u IT sektoru ($n = 21$). Uporedni prikaz srednjih ocena važnosti faktora dat je na slici 4.



Slika 4: Uporedni prikaz važnosti faktora za grupe: STUDENTI i ZAPOSLENI

Obe grupe ocenjuju *brzinu i odziv* kao najvažniji faktor, sa gotovo identičnim prosečnim ocenama. Studenti pridaju veću važnost *jednostavnosti korišćenja*, dok zaposleni više vrednuju *vizuelni izgled i UX*. Zaposleni takođe pokazuju veću otvorenost ka *praćenju AI trendova*, dok studenti daju blagu prednost *univerzalnosti alata* u odnosu na dostupnost dodataka (plugin-ova).

4.6 Analiza otvorenih odgovora ispitanika

Pored kvantitativnih podataka, ispitanici su pružili detaljnije obrazloženje svojih preferencija kroz otvorena pitanja o razlozima izbora glavnog alata i problemima sa kojima se susreću. Analiza ovih odgovora otkriva nekoliko ključnih obrazaca:

- Razlozi za izbor alata:** Dominantan faktor kod korisnika *Visual Studio Code-a* je balans između jednostavnosti i funkcionalnosti, pri čemu se često ističu intuitivnost i dobar korisnički doživljaj (UX). S druge strane, korisnici *JetBrains* porodice (IntelliJ, PyCharm, CLion) primarno biraju ove alate zbog robusne podrške za specifične

programske jezike poput Java i Spring framework-a. Zanimljivo je da značajan broj ispitanika kao razlog navodi eksterne faktore poput zahteva fakulteta ili radnog mesta („nametnuo posao”, „moram zbog faksa”), dok napredni korisnici (npr. *NeoVim*) fokus stavlaju na brzinu i efikasnost kroz eliminaciju upotrebe miša.

- **Problemi i ograničenja:** Kritike su jasno podeljene prema tipu alata. Kod kompleksnijih okruženja poput *Visual Studio*-a i *IntelliJ*-a, korisnici se najviše žale na veliku potrošnju memorije (RAM) i prisustvo „nepotrebnih opcija” koje opterećuju rad. Nasuprot tome, korisnici lakših editora (VSC) uočavaju nedostatke u naprednim funkcijama, kao što su slabija signalizacija grešaka u poređenju sa punim IDE rešenjima, problemi sa C++ podrškom, ili povremeni otkaz opcije „Go to Definition”.
- **Uloga pomoćnih alata i AI:** Iako su pitanja bila usmerena na razvojna okruženja, veliki broj ispitanika je spontano izdvojio *Git* kao svoj najkorišćeniji alat, definišući ga kao „apsolutno neophodan”. Takođe, primećuje se prodror novih alata poput *Cursor*-a, gde korisnici ističu visoku produktivnost, ali i uočavaju rane probleme poput AI halucinacija.

Analiza otvorenih odgovora pokazuje da se izbor razvojnog alata zasniva na balansu između funkcionalnosti, performansi i navike korisnika. Ispitanici su spremni da prihvate veću složenost ili hardversku zahtevnost alata ukoliko on donosi značajnu uštedu vremena i olakšava rad na kompleksnim projektima, dok se potencijalni pravci unapređenja uglavnom odnose na optimizaciju performansi, unapređenje signalizacije grešaka i bolje prilagođavanje specifičnim oblastima primene.

5 Diskusija

U ovom radu su analizirani obrasci korišćenja razvojnih alata među studentima tehničkih fakulteta i profesionalcima u IT sektoru, sa posebnim fokusom na kriterijume izbora i spremnost na promenu alata. Dobijeni rezultati se u velikoj meri poklapaju sa literaturom, ali ukazuju i na određene specifičnosti konteksta.

5.1 Poređenje dobijenih rezultata sa literaturom

Analizirani su najkorišćeniji razvojni alati među ispitanicima, a dobijeni rezultati su upoređeni sa relevantnom literaturom kako bi se sagledala njihova usklađenost sa opštim industrijskim trendovima.

- **IDE alati:** Među našim ispitanicima je najkorišćeniji alat *Visual Studio/Visual Studio Code*, što je u skladu sa literaturom u kojoj se ovaj alat nalazi na **prvom mestu** liste najkorišćenijih razvojnih okruženja [4]. Pored toga, razvojna okruženja iz *JetBrains* porodice imaju izraženu zastupljenost u našem uzorku, gde *IntelliJ IDEA* koristi 46,5% ispitanika, *PyCharm* 11,6%, a *CLion* 14%. Ovakva zastupljenost je takođe u saglasnosti sa literaturom, prema kojoj se *JetBrains IDEs* nalaze na **četvrtoj poziciji** najkorišćenijih razvojnih alata [4].
- **Tekstualni editori:** Najzastupljeniji tekstualni editori kod naših ispitanika su *Notepad++* (32,6%), *Vim* (27,9%) i *Kate* (25,6%), dok

Sublime Text koristi svega 9,3% ispitanika. Ovo predstavlja značajno odstupanje u odnosu na literaturu, u kojoj se *Sublime Text* nalazi čak na **drugom mestu** liste najkorišćenijih razvojnih alata. [4] Uočena razlika može se objasniti uticajem akademskog okruženja i ličnih navika korisnika, kao i relativno malim obimom uzorka.

- **Sistemi za verzionisanje koda:** Kada je u pitanju verzionisanje koda, *Git* se izdvaja kao ubedljivo najpopularniji sistem, kako prema literaturi, tako i u našem istraživanju, gde ga koristi čak 100% ispitanika [2]. Ovakav rezultat potvrđuje status *Git-a* kao industrijskog standarda u savremenom razvoju softvera, nezavisno od korišćenog razvojnog okruženja ili nivoa iskustva programera. U našoj anketi se jasno vidi znatna **dominacija GitHub-a nad GitLab-om** kao platformama za upravljanje verzijama koda — GitHub koristi 88,4% ispitanika, dok GitLab koristi 11,6%. Ovaj rezultat ukazuje na snažnu preferencu prema GitHub-u i u našem uzorku, što se poklapa sa opštim trendom u IT industriji, gde GitHub generalno ima znatno veći udeo u korišćenju u odnosu na druge Git platforme [5].

5.2 Faktori koji utiču na izbor alata

Rezultati istraživanja pokazuju da su **brzina i odziv razvojnog alata dominantni kriterijum izbora**, nezavisno od pola, radnog statusa ili nivoa iskustva ispitanika. Ovo potvrđuje da programeri primarno vrednuju efikasnost i stabilnost, budući da alati predstavljaju centralni deo njihovog svakodnevnog rada.

Analiza spremnosti na promenu alata ukazuje da ispitanici sa većim radnim iskustvom češće menjaju svoje primarno razvojno okruženje, što sugerise veću fleksibilnost i manju vezanost za konkretnе tehnologije. Nasuprot tome, manje iskusni programeri češće ostaju pri jednom alatu, verovatno usled formiranja početnih navika i potrebe za stabilnim okruženjem tokom učenja.

Razlike su uočene i u odnosu prema savremenim tehnologijama: zaposleni programeri pokazuju veću otvorenost ka praćenju *AI trendova* i korišćenju AI alata, dok studenti ove faktore ređe percipiraju kao važne pri izboru razvojnog okruženja.

Poseban značaj ima i **uticaj dugogodišnjih navika na izbor razvojnog alata**. Većina ispitanika je ocenila ovaj faktor veoma visoko — čak 62,8% ispitanika dalo je ocenu 4, a dodatnih 23,3% ocenu 5, dok zanemarljiv broj smatra da navike imaju mali ili nikakav uticaj. Ovakav rezultat ukazuje da se programeri često zadržavaju na alatima koje dobro poznaju, čak i kada postoje alternative sa sličnim ili naprednjim funkcionalnostima.

U celini, izbor razvojnog alata predstavlja rezultat kombinacije tehničkih zahteva, profesionalnog konteksta i stečenih navika, a ne isključivo procene pojedinačnih karakteristika alata.

5.3 Ograničenja istraživanja

Iako dobijeni rezultati pružaju uvid u aktuelne trendove korišćenja razvojnih alata i preference korisnika, istraživanje ima nekoliko ograničenja:

- **Veličina uzorka:** Uzorak od 43 ispitanika nije dovoljno obiman da bi se izvukli generalni zaključci za šиру populaciju programera.

- **Struktura uzorka:** Većina ispitanika su studenti Matematičkog fakulteta, što može dovesti do pristrasnosti prema određenim alatima.
- **Samoprocena ispitanika:** Podaci su zasnovani na subjektivnim procenama, koje mogu biti pod uticajem ličnih navika i trenutnog iskustva.
- **Ograničen skup faktora:** Nisu obuhvaćeni faktori poput specifičnih zahteva projekata, industrijskih standarda kompanija ili regionalnih razlika.

Uzimajući sve ovo u obzir, rezultati treba interpretirati sa oprezom i kao okvirne smernice, a ne kao definitivan prikaz navika i preferencija svih programera.

6 Zaključak

Na osnovu sprovedene ankete i analize dobijenih rezultata, može se zaključiti da su ***Visual Studio*** i ***Visual Studio Code*** dominantni alati u praksi, što je u skladu sa savremenim industrijskim trendovima i relevantnom literaturom.

Istraživanje je pokazalo da su **performanse alata, odnosno brzina i odziv, ključni kriterijumi** pri izboru razvojnog okruženja, dok faktori poput vizuelnog izgleda i praćenja AI trendova imaju manji, ali ne zanemarljiv značaj. Uočene razlike između studenata i zaposlenih ukazuju na promenu prioriteta tokom profesionalnog razvoja.

Rezultati takođe potvrđuju **dominaciju sistema *Git*** kao standarda za verzionisanje koda, kao i znatno veću popularnost platforme *GitHub* u odnosu na *GitLab*. Ovакви nalazi dodatno naglašavaju homogenost savremenih razvojnih praksi i oslanjanje na globalno prihvaćene alate i platforme.

Iako istraživanje ima određena ograničenja, pre svega u pogledu veličine i strukture uzorka, dobijeni podaci pružaju relevantan uvid u aktuelne navike i preferencije programera. Ovaj rad može poslužiti kao osnova za buduća, obimnija istraživanja koja bi uključila veći broj ispitanika, širi spektar tehnologija i dublju analizu uticaja novih trendova, poput veštačke inteligencije, na izbor razvojnih alata.

Literatura

- [1] Tara Dwyer. The Pros and Cons of Using a Code Editor vs. Integrated Development Environment (IDE), 2023. on-line at: <https://medium.com/@taradwyer17/the-pros-and-cons-of-using-a-code-editor-vs-integrated-development-environment-ide-643f151>
- [2] Geeks for geeks. Version Control Systems, 2025. on-line at: <https://www.geeksforgeeks.org/git/version-control-systems/>.
- [3] H. Hajjdiab I. Zayour. How Much Integrated Development Environments (IDEs) Improve Productivity? *Journal of Software*, 2013.
- [4] Mariusz Michalowski. Top 27 Software Development Tools and Platforms (2025 List), 2025. on-line at: <https://spacelift.io/blog/software-development-tools>.
- [5] Salman Ravoo. GitLab vs GitHub, 2025. on-line at: <https://kinsta.com/blog/gitlab-vs-github/>.

- [6] E. Sink. *Version Control by Example*. Pyrenean Gold Press, Boston, 2011.
- [7] I. Sommerville. *Software Engineering*. Addison-Wesley, 115 North Neil Street, Suite 408 Champaign, Illinois 61820, 2011.