

Najkorisniji razvojni alati

Seminarski rad u okviru kursa

Metodologija stručnog i naučnog rada

Matematički fakultet

Staša Đorđević

Lazar Savić

Đurđa Milošević

Bogdan Tomić

18. decembar 2025.

Sažetak

Analizirano je korišćenje razvojnih alata u oblasti programiranja, sa ciljem da se identifikuju faktori koji utiču na njihov izbor i eventualnu promenu. Istraživanje je sprovedeno putem anonimne ankete, kojom su obuhvaćeni demografski i profesionalni podaci ispitanika, njihove navike u radu sa razvojnim alatima, kao i stavovi o značaju različitih karakteristika alata. Dobijeni rezultati pružaju uvid u aktuelne trendove u upotrebi razvojnih okruženja i sistema za verzionisanje, kao i u razloge preferiranja određenih alata, čime se doprinosi boljem razumevanju procesa izbora razvojnog alata u savremenom programerskom okruženju.

Sadržaj

1	Uvod	3
2	O razvojnim alatima	3
2.1	Integrисана razvojna okruženja (IDE)	3
2.2	Tekstualni editori koda	4
2.3	Sistemi za verzionisanje	4
3	Metodologija	4
3.1	Uzorkovanje	4
3.2	Struktura uzorka	4
3.3	Struktura ankete	5
4	Rezultati	5
4.1	Analiza faktora važnosti pri izboru alata	6
4.2	Analiza korišćenja razvojnih okruženja u odnosu na pol	7
4.3	Analiza opštih preferencija Git platformi	8
4.4	Analiza uticaja radnog iskustva na promenu razvojnog alata	9
4.5	Analiza uticaja navike na promenu razvojnog alata	10
4.6	Analiza važnosti faktora kod korisnika tekstualnih editora .	11
4.7	Uporedna analiza prioriteta: Studenti nasuprot zaposlenih .	12
4.8	Analiza ograničenja razvojnih alata na osnovu otvorenih odgovora	13

5 Diskusija	14
5.1 Najkorisćeniji razvojni alati	14
5.2 Faktori koji utiču na izbor alata	15
5.3 Ograničenja istraživanja	16
6 Zaključak	16
Literatura	16

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 podrazumeva upotrebu različitih alata koji programerima omogućavaju efikasnije pisanje, organizaciju, testiranje i održavanje programskog koda. 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. Tipične komponente IDE-a uključuju editor koda, kompajler ili interpreter, alate za dinamičku analizu koda, kao i dodatne funkcionalnosti za upravljanje projektima i testiranje.

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.

Primeri integrisanih razvojnih okruženja koji se često koriste u praksi uključuju Visual Studio, IntelliJ IDEA, PyCharm i Android Studio.

Sa druge strane, integrisana razvojna okruženja mogu zahtevati veće hardverske resurse i biti složenija za prilagođavanje individualnim potrebama korisnika. Uprkos tome, njihova sveobuhvatnost čini ih posebno pogodnim kako za profesionalno okruženje, tako i za obrazovne svrhe.

2.2 Tekstualni editori koda

Tekstualni editori koda predstavljaju alate namenjene pisanju i uređivanju izvornog koda, sa fokusom na jednostavnost i fleksibilnost. 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. Ovakav modularni pristup korisnicima pruža veću kontrolu nad razvojnim okruženjem.

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 programskega koda, odnosno za upravljanje različitim verzijama softvera koji se razvija. Njihova osnovna uloga jeste čuvanje istorije izmena, omogućavanje povratka na prethodne verzije koda i olakšavanje saradnje više korisnika na istom projektu.

Upotrebojem sistema za verzionisanje, programeri mogu paralelno raditi na različitim delovima projekta, uz kasnije objedinjavanje izmena. Ovi sistemi značajno smanjuju rizik od gubitka podataka i doprinose boljoj timskoj organizaciji.

Sistemi za verzionisanje mogu se podeliti 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.

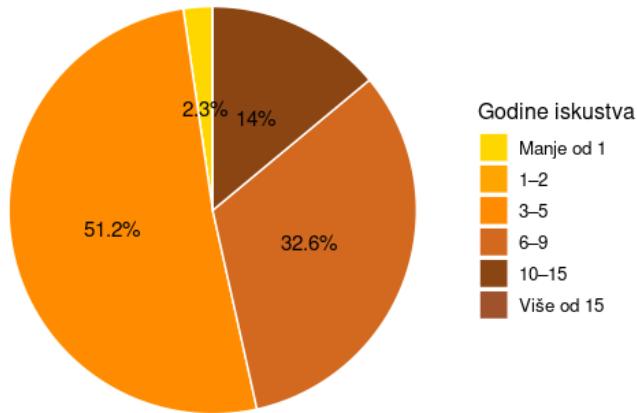
3.1 Uzorkovanje

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 su ispitanici dobrovoljno učestvovali u anketi i bili dostupni autorima istraživanja putem onlajn kanala.

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. Pregled ostalih demografskih karakteristika prikazan je na slikama 1 i 2.



Slika 1: Raspodela ispitanika po dužini iskustva u programiranju.

3.3 Struktura ankete

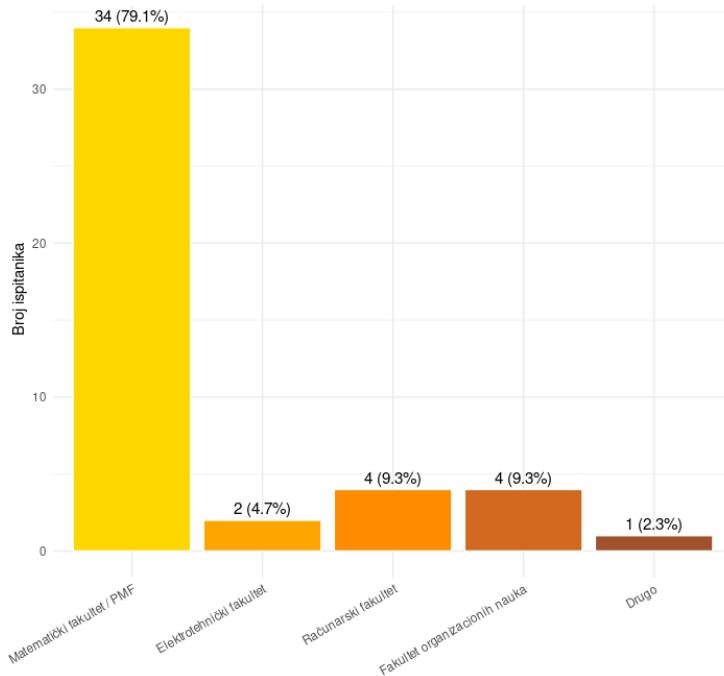
Anketa je sadržala 16 pitanja, od kojih je 12 bilo zatvorenog tipa, dok su četiri pitanja omogućavala slobodne odgovore. Pitanja su organizovana po celinama:

- **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, 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

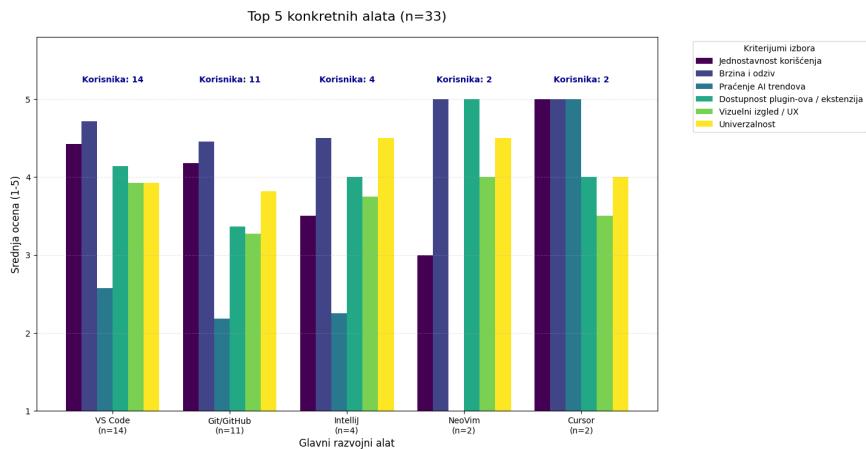
U ovom poglavljiju analizirani su podaci prikupljeni istraživanjem, uz poseban osvrт na korelacije između različitih parametara. Dobijeni uvid omogućava precizno definisanje trenutno najzastupljenijih i najefikasnijih razvojnih rešenja. Budući da uzorak obuhvata učesnike različitih nivoa stručnosti i obrazovnih profila, stanje na tržištu je sagledano iz više perspektiva. Vizuelni prikazi generisani su direktno iz prikupljenih odgovora, dok se prateća diskusija fokusira na prepoznavanje trendova i značajnih povezanosti unutar dobijenog skupa podataka.



Slika 2: Broj ispitanika po fakultetima.

4.1 Analiza faktora važnosti pri izboru alata

U ovom delu rada fokusiramo se na kriterijume koje programeri smatraju presudnim prilikom odabira radnog okruženja. Na slici 3 prikazani su rezultati za pet najzastupljenijih alata.



Slika 3: 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:

$$P_A = \left(\frac{f_{jednost} + f_{brzina} + f_{AI} + f_{plugin} + f_{visual} + f_{univ}}{6} \right) \cdot \frac{n_A}{N_{top5}} \quad (1)$$

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

Tabela 1: Opis parametara formule za reprezentativnost alata

Parametar	Opis faktora / Vrednost
$f_{jednost}$	Jednostavnost korišćenja
f_{brzina}	Brzina i odziv okruženja
f_{AI}	Praćenje i integracija AI trendova
f_{plugin}	Dostupnost dodataka i ekstenzija
f_{visual}	Vizuelni izgled i UX dizajn
f_{univ}	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{X}) i ključne karakteristike za svaki od analiziranih alata.

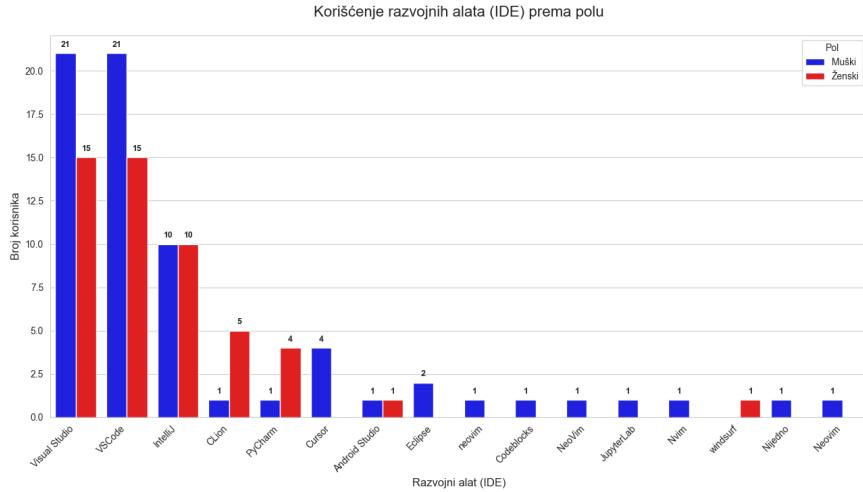
Tabela 2: Analiza vodećih razvojnih alata na osnovu pojedinačnih faktora važnosti

Alat	n	f_{jed}	f_{brz}	f_{AI}	f_{plg}	f_{vis}	f_{uni}	P_A
VS Code	14	4.4	4.7	2.6	4.1	3.9	3.9	4.10
Git/GitHub	11	4.2	4.5	2.2	3.4	3.3	3.8	3.70
IntelliJ	4	3.5	4.5	2.3	4.0	3.8	4.5	3.75
Cursor	2	5.0	5.0	5.0	4.0	5.0	5.0	4.83
NeoVim	2	3.0	5.0	2.5	5.0	4.0	4.5	4.25

Ovakav tabelarni prikaz jasno ukazuje na korelaciju između modernih zahteva (AI i brzina) i visokih ocena korisnika, dok VS Code zadržava primat u masovnosti primene.

4.2 Analiza korišćenja razvojnih okruženja u odnosu na pol

Ova podsekcija analizira distribuciju najpopularnijih razvojnih alata (IDE) među muškim i ženskim ispitanicima. Podaci su prikupljeni kako bi se utvrdilo da li postoji značajna korelacija između pola i izbora softvera za programiranje.



Slika 4: Korišćenje IDE alata u zavisnosti od pola ispitanika

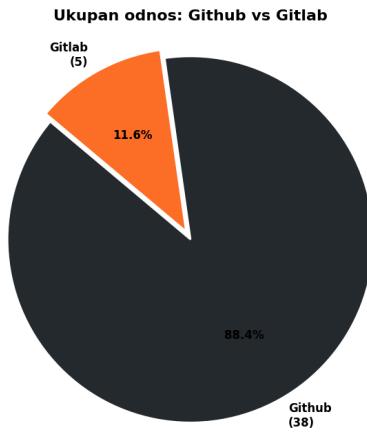
Na osnovu podataka prikazanih na slici 4, uočavamo odnos razvojnih alata u odnosu na pol ispitanika:

- **Dominantni alati:** *Visual Studio* i *VS Code* su najzastupljeniji alati u uzorku. Interesantno je da oba rešenja imaju identičnu strukturu korisnika, po 21 muških i 15 ženskih ispitanika, što ukazuje na njihovu široku prihvaćenost bez obzira na pol.
- **Uravnotežena zastupljenost:** Alat *IntelliJ IDEA* izdvaja se po potpunoj ravnopravnosti, sa tačno 10 muških i 10 ženskih korisnika, što ga čini statistički najneutralnijim alatom u pogledu preferencija polova.
- **Specifične razlike:**
 - Alati poput *CLion-a* i *PyCharm-a* pokazuju veću zastupljenost kod ženskih ispitanika u odnosu na njihov ukupan broj korisnika.
 - S druge strane, rešenja kao što su *Cursor* i *Eclipse* u ovom istraživanju su koristili isključivo muški ispitanici.

Ovi rezultati sugerisu da su industrijski standardi, poput Microsoft-ove porodice alata, univerzalno prihvaćeni u celoj zajednici. Razlike u preferencijama postaju uočljivije kod alata koji su usko specijalizovani za određene programske jezike ili kod novijih rešenja baziranih na veštačkoj inteligenciji.

4.3 Analiza opštih preferencija Git platformi

U okviru istraživanja, jedan od primarnih ciljeva bio je utvrđivanje zastupljenosti vodećih platformi za upravljanje verzijama koda (Git). Analiza je sprovedena na celokupnom uzorku ispitanika kako bi se dobio objektivan uvid u tržišni ideo i korisničke preferencije između dva najpopularnija rešenja: *GitHub-a* i *GitLab-a*. Distribucija ovih glasova prikazana je na slici 5.



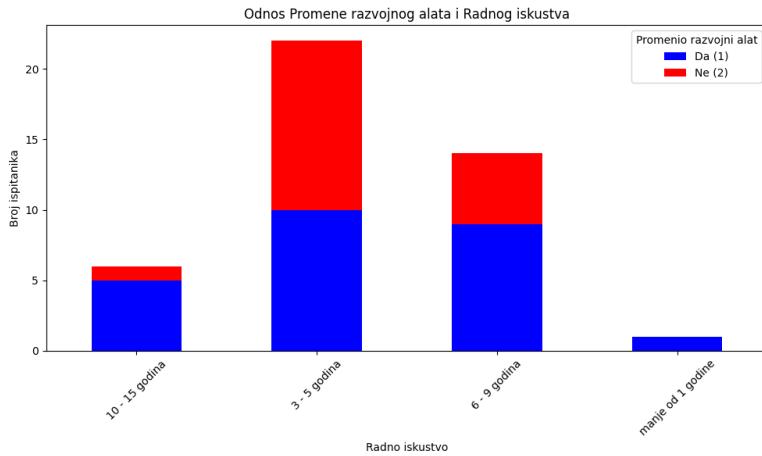
Slika 5: Ukupan odnos GitHub i GitLab korisnika na nivou celog uzorka

Na osnovu vizuelnog prikaza sa grafikona (slika 5), uočavaju se jasni trendovi i dominantna pozicija jednog rešenja u okviru ispitane populacije:

- **Apsolutna dominacija GitHub platforme:** Najveći deo uzorka, tačnije 88,4% (ukupno 38 ispitanika), opredelio se za *GitHub*. Ovako visok procenat potvrđuje status ove platforme kao globalnog standarda, što je verovatno uslovljeno širokom integracijom sa razvojnim alatima i primatom u *open-source* zajednici.
- **Zastupljenost GitLab-a:** Znatno manji deo ispitanika, odnosno 11,6% (ukupno 5 ispitanika), koristi *GitLab* kao primarnu platformu. Iako je deo korisnika manji, on predstavlja stabilan segment zajednice koji se verovatno oslanja na specifične *self-hosting* opcije ili napredne *CI/CD* funkcionalnosti po kojima je ovaj alat prepoznatljiv.
- **Zaključak o standardizaciji alata:** Dobijeni rezultati na ukupnom nivou pokazuju izrazitu homogenost u pogledu korišćenja pratećih tehnologija. 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.

4.4 Analiza uticaja radnog iskustva na promenu razvojnog alata

U ovom delu istraživanja analizira se korelacija između dužine radnog staža ispitanika i njihove tendencije ka promeni primarnog razvojnog alata. Rezultati prikazani na slici 6 pružaju uvid u to koliko su programeri u različitim fazama karijere skloni usvajanju novih tehnologija.



Slika 6: Uticaj radnog iskustva na promenu razvojnog alata

Na osnovu podataka prikazanih na slici 6, analizirana je spremnost ispitanika da promene svoj primarni razvojni alat u zavisnosti od godina radnog iskustva. Rezultati ukazuju na sledeće trendove:

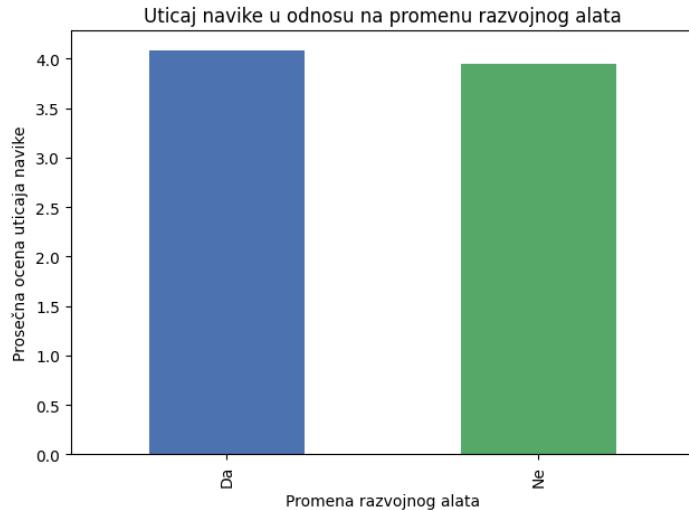
- **Iskusni programeri (10–15 godina):** Ova grupa pokazuje izuzetno visoku stopu promene alata. Od ukupno 6 ispitanika, čak 5 je prešlo na novo okruženje, što sugerije da iskusniji kadar ne okleva da usvoji modernija rešenja ukoliko ona nude veću efikasnost u radu.
- **Grupa sa 6–9 godina iskustva:** Kod ovih ispitanika je takođe uočen visok stepen prilagodavanja, gde je 9 od 14 programera promenilo svoj primarni alat.
- **Srednji nivo iskustva (3–5 godina):** Ovo je najbrojnija grupa u uzorku, u kojoj je uočljiva najveća podeljenost u navikama. Od 22 ispitanika, 10 je promenilo alat, dok je 12 ostalo pri svom prvobitnom izboru.
- **Početnici (manje od 1 godine):** Iako je uzorak u ovoj kategoriji minimalan, zabeleženo je da se promena alata dešava čak i u najranijoj fazi karijere.

Ukupni podaci ukazuju na to da programeri sa više radnog staža (preko 6 godina) zapravo češće menjaju alate nego što se to možda intuitivno očekuje. Razlog za ovo verovatno leži u evoluciji projekata na kojima rade, kao i u pojavi naprednjih alata baziranih na novim tehnologijama poput veštačke inteligencije.

4.5 Analiza uticaja navike na promenu razvojnog alata

Jedan od ciljeva istraživanja bio je ispitivanje uloge navike u procesu donošenja odluke o promeni glavnog razvojnog alata. U tom kontekstu analizirana je povezanost između ocene značaja navike odgovora ispitanika na pitanje da li su u prethodnom periodu menjali svoj primarni razvojni alat.

Na slici 7 prikazan je odnos prosečne ocene značaja navike između ispitanika koji su menjali glavni razvojni alat i onih koji su ostali pri postojećem alatu.



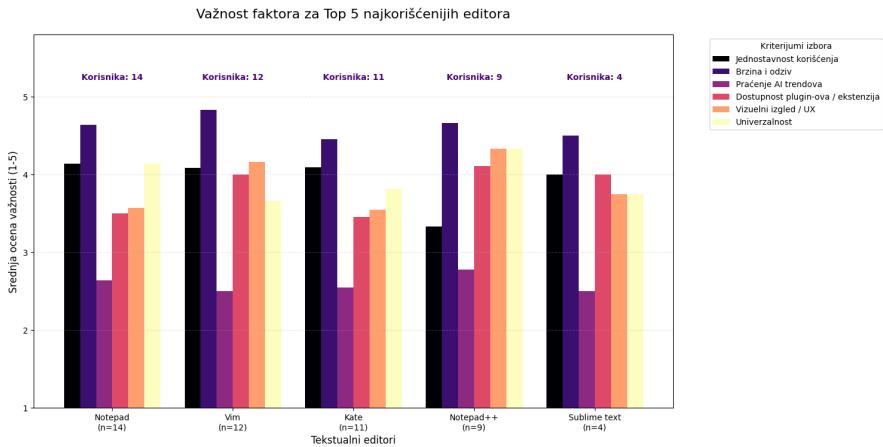
Slika 7: Uticaj navike na promenu razvojnog alata

Rezultati pokazuju da obe grupe ispitanika naviku ocenjuju kao važan faktor u izboru razvojnog alata. Iako je početna pretpostavka bila da će ispitanici kojima navika ima manji značaj biti skloniji promeni alata, uočava se da i ispitanici koji su menjali alat daju relativno visoke ocene uticaju navike. Razlika između prosečnih vrednosti ove dve grupe je prisutna, ali nije izražena, što ukazuje na to da navika ima snažan, ali ne i presudan uticaj na odluku o promeni razvojnog alata.

Dobijeni rezultati sugeriraju da promena razvojnog alata ne podrazumeva odsustvo navike, već pre spremnost korisnika da postojeće navike prevaziđu u situacijama kada novi alat nudi jasne funkcionalne prednosti ili bolje odgovara zahtevima konkretnog projekta.

4.6 Analiza važnosti faktora kod korisnika tekstualnih editora

Pored kompleksnih integrisanih razvojnih okruženja (IDE), značajan deo ispitanika oslanja se na tekstualne editore u svakodnevnom radu. Na slici 8 prikazano je kako korisnici pet najzastupljenijih editora ocenjuju ključne faktore prilikom izbora svog primarnog alata.



Slika 8: Važnost faktora u odnosu na najčešće korišćene tekstualne editore

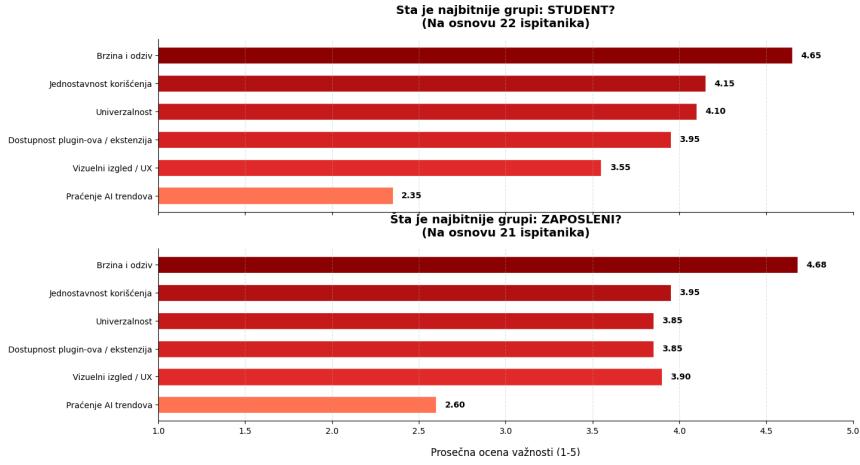
Na osnovu prikupljenih odgovora, analizirane su prosečne vrednosti šest ključnih kriterijuma: brzina i odziv, jednostavnost korišćenja, univerzalnost, dostupnost dodataka, vizuelni izgled i AI trendovi. Podaci sa slike 8 ukazuju na specifične prioritete unutar različitih korisničkih grupa:

- **Notepad** ($n = 14$): Najzastupljeniji editor u uzorku. Korisnici mu daju visoke ocene za brzinu i odziv, kao i za jednostavnost korišćenja, dok su AI trendovi očekivano najslabije ocenjen faktor.
- **Vim** ($n = 12$): Ovaj alat beleži izuzetno visoku ocenu za brzinu i odziv (blizu 5.0). Korisnici takođe visoko vrednuju dostupnost dodataka i vizuelni izgled/UX, što ukazuje na visok stepen personalizacije.
- **Kate** ($n = 11$): Karakteriše ga ujednačenost ocena, sa posebnim naglaskom na brzinu rada. Interesantno je da korisnici ovog editora daju stabilne ocene univerzalnosti alata.
- **Notepad++** ($n = 9$): Kod ovog alata dominiraju ocene za brzinu, vizuelni izgled i univerzalnost, što ga pozicionira kao svestrano rešenje za brze izmene koda.
- **Sublime Text** ($n = 4$): Iako na manjem uzorku, ovaj editor zadržava visok nivo ocena za brzinu i jednostavnost, dokazujući svoju reputaciju jednog od najbržih modernih tekstualnih editora.

Ova analiza potvrđuje da korisnici tekstualnih editora absolutni prioritet daju performansama i brzini 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.

4.7 Uporedna analiza prioriteta: Studenti nasu- prot zaposlenih

Finalni segment analize fokusira se na razlike u prioritetima između studenata ($n = 22$) i zaposlenih u IT sektoru ($n = 21$). Poređenje rezultata na objedinjenom prikazu (slika 9) omogućava precizan uvid u evoluciju potreba programera tokom profesionalnog razvoja.



Slika 9: Uporedni prikaz važnosti faktora za grupe: STUDENTI i ZAPOSLENI

Na osnovu uporednog pregleda srednjih ocena važnosti (\bar{X}) sa slike 9, uočavaju se sledeće ključne razlike:

- **Neprikosnovenost performansi:** Obe grupe su *Brzinu i odziv* ocenile kao najvažniji faktor. Zaposleni daju nijansu višu ocenu (4.68) u odnosu na studente (4.65), što potvrđuje da je efikasnost alata absolutni prioritet bez obzira na profesionalni status.
- **Jednostavnost nasuprot profesionalnom UX-u:**
 - **Studenti** daju značajno veći prioritet *Jednostavnosti korišćenja* (4.15), što reflektuje potrebu za brzim savladavanjem alata tokom procesa učenja.
 - **Zaposleni** pokazuju blagi pad važnosti puke jednostavnosti (3.95), ali zato daju primetno veću važnost faktoru *Vizuelni izgled / UX* (3.90 naspram studentskih 3.55). Ovo ukazuje na to da profesionalci, usled dugotrajnog svakodnevnog rada u alatima, više vrednuju ergonomiju i kvalitet dizajna interfejsa.
- **Ekstenzibilnost i univerzalnost:** Kod zaposlenih je uočena potpuna ujednačenost u vrednovanju *Univerzalnosti* i *Dostupnosti plugin-ova* (obe ocene su 3.85), dok studenti daju blagu prednost univerzalnosti alata (4.10) u odnosu na ekstenzije (3.95).
- **Odnos prema AI inovacijama:** Profesionalni sektor pokazuje veću otvorenost ka novim tehnologijama, pa **zaposleni** daju višu ocenu *Praćenju AI trendova* (2.60) u odnosu na **studente** (2.35). Ovo sugerira da iskusniji programeri brže prepoznaju upotrebnu vrednost AI asistenata u realnim produkcionim uslovima.

Može se zaključiti da, dok su bazični zahtevi za brzinom rada univerzalni, profesionalizacija donosi povećan fokus na korisničko iskustvo (UX) i AI integracije, dok važnost same jednostavnosti primene blago opada u korist naprednijih i bolje dizajniranih funkcionalnosti.

4.8 Analiza ograničenja razvojnih alata na osnovu otvorenih odgovora

U cilju dubljeg razumevanja uočenih problema u svakodnevnoj upotrebni razvojnih alata, analizirani su odgovori ispitanika na otvoreno pitanje

koje se odnosilo na ograničenja i nedostatke alata koje najčešće koriste. Odgovori su povezani sa glavnim razvojnim alatom koji su ispitanici naveli i zatim analizirani primenom tematske analize.

Kod korisnika alata Visual Studio Code najčešće su istaknuti problemi u vezi sa performansama i potrošnjom sistemskih resursa, kao i ograničenja u podršci za programski jezik C++. Ispitanici navode slabiju signalizaciju grešaka, nedovoljno pouzdane funkcije navigacije kroz kod, kao i tendenciju ka uvođenju funkcionalnosti koje ne doprinose direktno produktivnosti, ali povećavaju opterećenje sistema.

Korisnici integriranog razvojnog okruženja IntelliJ IDEA ističu suprotni problem, odnosno preveliki broj opcija i funkcionalnosti koje mogu otežati snalaženje i rad, naročito u slučajevima kada korisnicima nije potrebna kompletna paleta naprednih mogućnosti.

Kod korisnika tekstualnih editora, posebno NeoVim-a, kao glavno ograničenje navodi se složenost konfiguracije i potreba za dodatnim znanjem, budući da se prilagodavanje i razvoj ekstenzija zasniva na programskom jeziku Lua. Iako ovaj pristup pruža visok stepen fleksibilnosti, on istovremeno predstavlja prepreku za širu upotrebu.

Za alate poput Notepad++ uočena su ograničenja u domenu korisničkog interfejsa, koji se percipira kao zastareo u poređenju sa savremenim razvojnim okruženjima. Kod Jupyter okruženja ispitanici navode da ono nije dovoljno optimizovano za zahtevnije zadatke u oblasti analize podataka i mašinskog učenja, što ograničava njegovu primenu u kompleksnijim projektima.

Na osnovu identifikovanih problema mogu se izdvojiti potencijalni pravci unapređenja, koji uključuju optimizaciju performansi, unapređenje signalizacije grešaka, pojednostavljenje korisničkog interfejsa kao i bolje prilagodavanje alata 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 Najkorišćeniji razvojni alati

Analizirani su najkorišćeniji razvojni alati među ispitanicima, a rezultati su upoređeni sa relevantnom literaturom, uključujući izvore [1] za IDE i textualne editore, [2] za sisteme za verzionisanje, kao i [3] za platforme poput GitHub-a i GitLab-a.

- **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. 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.

- **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. 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. 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 preferenciju prema GitHub-u i u našem uzorku, što se poklapa sa opštim trendom u IT industriji, gde GitHub generalno ima znatno veći udio u korišćenju u odnosu na druge Git platforme.

5.2 Faktori koji utiču na izbor alata

Pored same zastupljenosti pojedinih alata, rezultati istraživanja pružaju uvid u obrasce ponašanja i navike programera prilikom izbora i promene razvojnog okruženja. Analiza faktora izbora pokazuje da su *brzina i odziv* alata dosledno ocenjeni kao najvažniji kriterijum, nezavisno od pola, radnog statusa ili nivoa iskustva ispitanika. Ovakav rezultat potvrđuje da programeri primarno vrednuju efikasnost i stabilnost, budući da razvojni alati predstavljaju centralni deo svakodnevnog rada.

Takođe, analiza spremnosti na promenu razvojnog alata ukazuje da programeri sa većim radnim iskustvom češće menjaju svoje primarno okruženje. Ovo sugerije da se sa razvojem karijere smanjuje vezanost za konkretni alat, dok raste fokus na prilagodljivost i efikasnost u različitim tehnološkim kontekstima. Nasuprot tome, manje iskusni programeri češće ostaju pri jednom alatu, što može biti posledica formiranja početnih navika i potrebe za stabilnim okruženjem tokom učenja.

Razlike su uočljive i u odnosu prema savremenim tehnologijama. Zaposleni programeri pokazuju veću otvorenost ka praćenju *AI trendova* i korišćenju AI alata u razvoju softvera, što može biti posledica rada na realnim produpcionim projektima i potrebe za većom produktivnošću. Studenti, iako upoznati sa ovim tehnologijama, u manjoj meri ih doživljavaju kao važan faktor pri izboru razvojnog alata.

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, rezultati istraživanja ukazuju da izbor razvojnog alata nije zasnovan isključivo na tehničkim karakteristikama, već predstavlja kombinaciju profesionalnih potreba, stecenih navika i konteksta u kojem programer radi ili se obrazuje.

5.3 Ograničenja istraživanja

Iako dobijeni rezultati pružaju uvid u aktuelne trendove korišćenja razvojnih alata i preference korisnika, istraživanje ima nekoliko ograničenja koja treba uzeti u obzir pri tumačenju rezultata.

Prvo, uzorak od 43 ispitanika nije dovoljno obiman da bi se izvukli generalni zaključci za šиру populaciju programera. Za značajnije statističke procene i pouzdano određivanje trendova, bio bi potreban znatno veći i raznovrsniji uzorak.

Drugo, sastav uzorka je specifičan — većina ispitanika su studenti Matematičkog fakulteta, što može povući pristrasnost prema određenim alatima, budući da se na fakultetu koriste uglavnom određeni razvojni alati i okruženja.

Treće, rezultati su zasnovani na samoproceni ispitanika, što može biti pod uticajem subjektivnih stavova, navika ili trenutnog iskustva. Navike korišćenja alata, prethodno iskustvo i lične preferencije mogu značajno uticati na odgovore, što ograničava mogućnost da se dobijeni podaci trebiraju kao apsolutno reprezentativni.

Konačno, anketa nije obuhvatila sve moguće faktore koji utiču na izbor alata, kao što su specifični zahtevi projekata, industrijski standardi u kompanijama ispitanika ili regionalne razlike, što znači da određene varijable koje mogu biti značajne nisu uzete u obzir.

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

Cilj ovog seminar skoga rada bio je da se ispita koji razvojni alati se najčešće koriste među studentima i profesionalcima u IT sektoru, kao i koji faktori imaju najveći uticaj na njihov izbor. 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*. Ovakvi 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] Top 27 Software Development Tools and Platforms (2025 List) at:
<https://spacelift.io/blog/software-development-tools>
- [2] Version Control Systems at:
<https://www.geeksforgeeks.org/git/version-control-systems/>
- [3] GitLab vs GitHub at:
<https://evbn.org/gitlab-vs-github-explore-their-major-differences-and-similarities-1678449>