

Dopunite naslov svoga rada
Dopunite autore rada

22. april 2019

Sadržaj

| | |
|---|-----------|
| 1 Uputstva | 2 |
| 2 Recenzent — ocena: | 3 |
| 2.1 O čemu rad govori? | 3 |
| 2.2 Krupne primedbe i sugestije | 3 |
| 2.3 Sitne primedbe | 3 |
| 2.4 Provera sadržajnosti i forme seminarskog rada | 4 |
| 2.5 Ocenite sebe | 6 |
| 3 Recenzent — ocena: | 7 |
| 3.1 O čemu rad govori? | 7 |
| 3.2 Krupne primedbe i sugestije | 7 |
| 3.3 Sitne primedbe | 7 |
| 3.4 Provera sadržajnosti i forme seminarskog rada | 8 |
| 3.5 Ocenite sebe | 9 |
| 4 Recenzent — ocena: | 10 |
| 4.1 O čemu rad govori? | 10 |
| 4.2 Krupne primedbe i sugestije | 10 |
| 4.3 Sitne primedbe | 10 |
| 4.4 Provera sadržajnosti i forme seminarskog rada | 10 |
| 4.5 Ocenite sebe | 11 |
| 5 Dodatne izmene | 12 |

Glava 1

Uputstva

Prilikom predavanja odgovora na recenziju, obrišite ovo poglavlje.

Neophodno je odgovoriti na sve zamerke koje su navedene u okviru recenzija. Svaki odgovor pišete u okviru okruženja \odgovor, [kako bi vaši odgovori bili lakše uočljivi](#).

1. Odgovor treba da sadrži na koji način ste izmenili rad da bi adresirali problem koji je recenzent naveo. Na primer, to može biti neka dodata rečenica ili dodat pasus. Ukoliko je u pitanju kraći tekst onda ga možete navesti direktno u ovom dokumentu, ukoliko je u pitanju duži tekst, onda navedete samo na kojoj strani i gde tačno se taj novi tekst nalazi. Ukoliko je izmenjeno ime nekog poglavlja, navedite na koji način je izmenjeno, i slično, u zavisnosti od izmena koje ste napravili.
2. Ukoliko ništa niste izmenili povodom neke zamerke, detaljno obrazložite zašto zahtev recenzenta nije uvažen.
3. Ukoliko ste napravili i neke izmene koje recenzenti nisu tražili, njih navedite u poslednjem poglavlju tj u poglavlju Dodatne izmene.

Za svakog recenzenta dodajte ocenu od 1 do 5 koja označava koliko vam je recenzija bila korisna, odnosno koliko vam je pomogla da unapredite rad. Ocena 1 označava da vam recenzija nije bila korisna, ocena 5 označava da vam je recenzija bila veoma korisna.

NAPOMENA: Recenzije ce biti ocenjene nezavisno od vaših ocena. Na osnovu recenzije ja znam da li je ona korisna ili ne, pa na taj način vama idu negativni poeni ukoliko kažete da je korisno nešto što nije korisno. Vašim kolegama šteti da kažete da im je recenzija korisna jer će misliti da su je dobro uradili, iako to zapravo nisu. Isto važi i na drugu stranu, tj nemojte reći da nije korisno ono što jeste korisno. Prema tome, trudite se da budete objektivni.

Glava 2

Recenzent — ocena:

2.1 O čemu rad govori?

U ovom radu je veoma kratko prikazan programski jezik Clojure, koji je pre svega funkcionalan i dinamičan jezik. Predstavlja se njegova povezanost sa jezicima Lisp i Java, uvode se novi pojmovi kao što su homoikoničnost, detaljno se opisuje sintaksna struktura i moguća okruženja za programski jezik.

2.2 Krupne primedbe i sugestije

U radu nedostaje postojanje povezane priče, odnosno tok upoznavanja jezika sa jednostavnog nivoa na viši na kojem može da se vidi snaga jezika. Nakon uvoda i zaključka postoji mnogo slobodnog mesta koje bi trebalo da se popuni. Takođe poglavlje o instalaciji programskog jezika je veoma detaljno objašnjeno, pa bi moglo da se skrati, kako bi se dodalo nešto interesantnije.

Sugestije koje mogu obogatiti kvalitet rada:

1. Kako se koriste tehnike objektno-orijentisanog programiranja u Clojure (na primer polimorfizam)
2. Kako se omogućava enkapsulacija ako jezik nije orijentisan oko klasa
3. Detaljnije o rekurziji i operatoru recur
4. Zašto makroi igraju važnu ulogu u jeziku
5. Kako imutabilni objekti doprinose moći konkurentnosti
6. Detaljnije o REPL

2.3 Sitne primedbe

1. Strana 1, naslov poglavlja Framework bi mogao da se prevede na srpski - okruženje

2. Korišćenje dva načina pisanja reči dijalekt: dijalekt (Uvod) i dijalekat (Poglavlje 3).
3. U radu se provlači na više mesta reč web, umesto veb
4. Strana 3, poglavlje 2, pasus 1: tačka na početku drugog reda, Lisp-olikih umesto lispolikih, trebalo bi da proverite značenje rečenice, pošto deluje kao da se ne uklapa kraj rečenice
5. Strana 4, poglavlje 3.1, pasusi 2 i 3: Java-om, Clojure-u, JavaScript-om, C++-om
6. Strana 7, poglavlje 5: treće korišćenje iste rečenice "Clojure je dijalekat programskog jezika Lisp"
7. Strana 8, poglavlje 5.1 (poslednji pasus), poglavlje 5.2 i 5.4 (pasus 1): Clojure-u
8. Strana 8, poglavlje 5.3, druga rečenica: fali predlog **na**
9. Strana 9, poglavlje 6, pasus 1: gotovo svakom JVM-u umesto svakoj JVM-i
10. Strana 10, poglavlje 6.1, pasus 1: neumesna upotreba "...laganih biblioteka..."
11. Strana 10, poglavlje 6.1, pasus 2: šablonskong, psotoji umesto šablonskog, postoji
12. Strana 10, poglavlje 6.2, pasus 2: frontend umesto front-end
13. Strana 10, poglavlje 6.3, pasus 1: backend, generirati umesto back-end, generisati
14. Strana 11, poglavlje 7, pasus 2: sa sve većim brojem platformi umesto na sve većem broju platformi

2.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?

Rad delimično odgovara na zadatu temu. Pokriveni su važni aspekti upoznavanja sa jezikom kao što je sintaksna struktura, ali trebalo bi da se dodaju konkretne specifičnosti jezika.

2. Da li je nešto važno propušteno?

U radu nisu prikazani primeri, koji bi dočarali moć jezika. Sugestija bi bila dodati jedno poglavlje sa prolaskom kroz primere.

3. Da li ima suštinskih grešaka i propusta?

Uvod u rad ne oslikava dobro sadržaj rada. Naime, spominju se velike mogućnosti koje pruža Clojure, ali se u radu to ne prikazuje. Takođe u

zaključku imamo zaključke koji se ne mogu doneti na osnovu ovog rada. Tačnije, priča se o tome kako je jezik elegantan i kako zahteva manje kodiranja od drugih jezika, što iz rada ne možemo videti jer nemamo odgovarajuće primere. Spominje se konkurentnost o kojoj u radu nije bilo reči.

Sugestije za bolje razumevanje rada:

- (a) Strana 3, poglavlje 2, pasus 2: Moglo bi da se doda koja konkretna zajednica je u pitanju
- (b) Ubačene slike u poglavju 2.2 prikazuju neki odnos jezika Clojure sa ostalim jezicima, ali bi moglo da se pojasni koji je to zapravo odnos.
- (c) Trebalo bi da se napiše jasnija definicija homoikoničnosti.
- (d) Strana 5, poglavlje 3.2, pasus 3: Moglo bi da se doda konkretno koje probleme donosi mutabilnost objekta i kako to rešava Clojure
- (e) Strana 7, poglavlje 5: Spominje se pojam uniformnost, trebalo bi da se doda objašnjenje tog pojma, pošto piše da je karakteristično za jezike kao što je Clojure
- (f) Strana 8, poglavlje 5.2: Mogla bi da se doda upotreba literala `nil`
- (g) Strana 10, poglavlje 6.1: Moglo bi da se objasni šta je mikro-framework
- (h) Strana 10, poglavlje 6.2: Trebalo bi da se pojasni prva rečenica, s obzirom da je nejasno šta znači "... idiosinkrazije veb platforme..."

4. Da li je naslov rada dobro izabran?

Za naslov rada sve pohvale, smatram da je jedan od boljih: kratak, interesantan i privlačan.

5. Da li sažetak sadrži prave podatke o radu?

Generalno da, spominje se da je jezik dinamičan, što je u radu i opisano. Takođe je spomenuto da je jezik sa naglaskom na funkcionalno programiranje, međutim se u radu jako malo priča o tome. Nakon čitanja rada, ne stiče se utisak poznavanja jezika dovoljno da čitaoc može da kaže koje su to oblasti i načini primene jezika. Ključna reč konkurentnost bi trebalo da se skloni, s obzirom da se u radu ne spominje konkurentnost.

6. Da li je rad lak-težak za čitanje?

Rad je lak za čitanje, ali jedna sugestija bi bila da se rečenice bez mnogo smisla i značaja sklone. Na primer rečenica u delu o razvoju programskog jezika: "Clojure konferencije, od kojih su najposećenije one u Sjedinjenim državama i evropske konferencije, organizuju se svake godine širom planete." Ovakva rečenica bi mogla da se skloni, s obzirom da se održava milion konferencija širom sveta za raznorazne jezike, Clojure nije izuzetak.

7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?

Za razumevanje teksta nije potrebno predznanje, mnogo se temeljno i detaljno objašnjavaju trivijalne stvari, dok teže stvari nisu pokrivene u radu. Moglo bi da se smanji detaljisanje jednostavnih stvari kao što je opis naredbe kontrole toka if i da se objasne neke teže i zanimljivije stvari specifične jeziku.

8. Da li je u radu navedena odgovarajuća literatura?

U radu je navedena odgovarajuća literatura, sadrži više od traženog minimuma.

9. Da li su u radu reference korektno navedene?

U prvom delu rada su korektno navedene reference, dok u poglavlju Framework nisu, trebalo bi da se ubace unutar rečenice. Mogla bi da se doda referenca u poglavlju 3.3 na dokumentaciju interfejsa REPL.

10. Da li je struktura rada adekvatna?

Struktura rada je adekvatna, sadrži sve što jedan seminarski rad treba da sadrži. Jedina sugestija bi bila da se poglavlje o instalaciji programskog jezika pomeri sa centralnog mesta u radu, na primer moglo bi da se pozicionira posle poglavlja Framework.

11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?

Da, rad sadrži sve elemente propisane uslovima.

12. Da li su slike i tabele funkcionalne i adekvatne?

Tabela 1 ispunjava većinu zahteva kao što su poravnanje u okviru tabele, nepostojanje redundantnosti, međutim naslov tabele bi trebalo da ide iznad tabele i ne bi trebalo da postoji referisanje u okviru naslova tabele. Mogli bi da se pojašne nepoznati termini koji se koriste u tabeli kao što su reduktori, transduktori u pasusu u okviru poglavlja. Trebalo bi da proverite da li je vizuelno preglednija tabela ukoliko se sklone vertikalne linije.

Slike su ispravno dodate u rad, međutim za sliku broj 1 ne bi trebalo da postoji referisanje u naslovu i ne postoji referisanje iz teksta na nju.

Za slike i za tabele (ako zanemarimo nepoznate pojmove) važi da se razumeju bez čitanja teksta.

2.5 Ocenite sebe

U ovu oblast sam malo upućena i to na osnovu literature koja se nudi na kraju rada, ali takođe sam pri istraživanju naišla na još jednu literaturu sa mnogo primera koja može pomoći: "The Joy of Clojure" M. Fogus, C. Houser.

Glava 3

Recenzent — ocena:

3.1 O čemu rad govori?

Autori su u ovom radu dali opis programskog jezika Clojure. Najpre je opisan nastanak jezika i predstavljeno je kako se jezik uklapa u programske paradigme kojima pripada, prvenstveno funkcionalnoj. Nakon kratkog uputstva o instalaciji jezika, sledi opis sintakse i par primera koda napisanih u Clojure-u s namerom da pokažu osnovne mogućnosti jezika. Na kraju dolazi opis tri framework-a za Clojure.

3.2 Krupne primedbe i sugestije

Uzimajući u obzir rad kao celinu, moje mišljenje je da je rad napisan na izuzetno visokom nivou i da sadržaj istog pokriva sve bitne informacije u vezi jezika, ne ulazeći previše u napredne aspekte. Rad je takođe bilo izuzetno lako pročitati i ući u tematiku bez ikakvog prethodnog predznanja o jeziku.

Postoji par krupnijih problema za koje bih predložio ispravku u sledećoj reviziji rada, a to su:

- Uputstvo za instalaciju na Linux platformama ne pokriva sve distribucije Linux-a, naime, u radu je prikazano kako instalirati Clojure na sistemima koji koriste *apt* kao sistem za upravljanje paketima. Bitno je to naglasiti u naslovu podsekcije ili pokriti i ostale distribucije Linux-a.
- U sekciji koja opisuje framework-ove za Clojure, postoji dosta anglicizama i skraćenica koje narušavaju čitljivost i razumljivost rada. Možda je bolje ostaviti idejni opis pomenutih framework-ova, bez opisa njihovih detalja.

3.3 Sitne primedbe

Sledi lista uočenih sitnih primedbi:

- Imena autora i adrese elektronske pošte nisu centrirane.
- U listi ključnih reči ne postoji Clojure koji je tema rada.

- Prva rečenica druge sekcije nije napisana kako treba.
- Referencu na drugu sliku možda treba staviti u zagradi kao (Slika 2). Ukoliko neko odštampa rad, neće imati znati da je to referenca na sliku.
- Podsekcija 3.2 na kraju spominje *recur* operator i naglo se završava, možda će kratak opis operatora pomoći.
- Podsekcija *Pokretanje*, pored opisa pokretanja, daje i opis korišćenja biblioteke, pa bih predložio promenu naslova.
- Tri puta je spomenuto da je Clojure dijalekat programskog jezika Lisp, možda je bolje u nekim slučajevima preformulisati taj izraz.
- U podsekciji 5.2 nakon listinga, kada se spominje *else* grana, treba formatirati ključnu reč *else* nalik ostalim.
- Podsekcija 6.1 sadrži jednu štamparsku grešku u drugom pasusu. (**pstoji**)
- U podsekciji 6.2 u poslednjoj rečenici postoji štamparska greška (**obezbeđujuć i**)

3.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
Da, rad daje kvalitetan i sadržajan opis date teme.
2. Da li je nešto važno propušteno?
Smatram da su pokrivene sve bitne karakteristike jednog programskog jezika.
3. Da li ima suštinskih grešaka i propusta?
U vezi same teme i srži rada, ne.
4. Da li je naslov rada dobro izabran?
Naslov rada ne pruža uvid u temu ali je originalan i privlači pažnju.
5. Da li sažetak sadrži prave podatke o radu?
Da, pruža dovoljno informacija o tematici i sadržaju rada.
6. Da li je rad lak-težak za čitanje?
Rad je u globalu izuzetno lak za čitanje. Jedina problematična sekcija jeste šesta.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
Potrebno je osnovno znanje iz oblasti programiranje i poznavanje nekih programskih paradigmi.
8. Da li je u radu navedena odgovarajuća literatura?
Literatura je korektna i sadržajna.
9. Da li su u radu reference korektno navedene?
Jesu.

10. Da li je struktura rada adekvatna?
Rad poštuje predloženu strukturu.
11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
Rad poštuje sve propisane elemente.
12. Da li su slike i tabele funkcionalne i adekvatne?
Jesu i prikazuju interesantne stvari koje je teže reći kroz tekst.

3.5 Ocenite sebe

d) malo upućeni

Sa jezikom Clojure nisam nikada radio, niti video neki primer koda do sada. Medjutim, poznajem koncepte na kojima počiva i jezike koji podržavaju integraciju Clojure-a.

Glava 4

Recenzent — ocena:

4.1 O čemu rad govori?

Upoznajemo se sa osnovnom idejom nastajanja Clojure-a. Funkcionalan programski jezik ciji kod se prevodi u Java bajt kod i izvršava se na JVM. Bitna osobina je homoikoničnost. Osnovni interfejs za programiranje je REPL - konzolni interfejs gde se Clojure kod evaluira za svaku napisanu liniju. Upućeni smo kako da instaliramo Clojure. Predstavljena su nam osnovna pravila sintakse (literali, operatori, kontrola toka, definisanje funkcija) i strukture podataka (liste, mape, skupovi). Opisani su neki od najpoznatijih Clojure framework-ova: Luminus, Hoplon i ClojureHomePage.

4.2 Krupne primedbe i sugestije

Kao neko ko nema predznanje iz ove oblasti, smatram da fali sekcija koja će me ubediti da baš ovaj programski jezik želim da naučim. Tj. bilo bi lepo da se izdvoji u par rečenica koja je glavna upotreba Clojure-a, koje su osnovne primene, u koje svrhe i koji tip problema je pogodno rešavati ovim jezikom.

4.3 Sitne primedbe

Rad je gramatički i stilski ispravno napisan. Jedina primedba se odnosi na spisak autora koji je ispao iz margina teksta, dosta je šire od ostatka rada.

4.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
Uz nedostatak koji sam naveo u sekciji za primedbe, smatram da je rad odgovorio na temu.
2. Da li je nešto važno propušteno?
Navedeno u primedbama.

3. Da li ima suštinskih grešaka i propusta?
Smatram da nema suštinskih propusta, tema sadrži uvod u materiju, objašnjenje materije kao i zaključak autora.
4. Da li je naslov rada dobro izabran?
Mislim da u naslovu rada mora stajati i ključna reč Clojure koja će unapred reći čitaocu o čemu je rad.
5. Da li sažetak sadrži prave podatke o radu?
Sažetak u potpunosti sadrži prave podatke o radu, s tim što ne bi bilo loše napisati isto to ali prostijim rečima, manje stručnih izraza, kako bi neko ko pročita sažetak imao želju da pročita ceo rad iako nije uopšte upućen u materiju o kojoj se piše, inače neće razumeti sažetak i odustaće.
6. Da li je rad lak-težak za čitanje?
Smatram da je rad lak za čitanje.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
Za razumevanje celokupnog rada nije potrebno veliko predznanje iz oblasti funkcionalnog programiranja. Cilj rada nije da čitalac nauči funkcionalno programiranje već da stvori širu sliku i da bude zainteresovan za oblast o kojoj čita. Iz osnova programiranja svakako da jeste potrebno što je i logično. Jedino u sažetku kao što napisah bi bilo lepo uprostiti ga.
8. Da li je u radu navedena odgovarajuća literatura?
U radu je navedena odgovarajuća literatura i precizno je označena.
9. Da li su u radu reference korektno navedene?
Sve reference su korektno navedene.
10. Da li je struktura rada adekvatna?
Struktura je adekvatna i tok kojim se piše je smislen.
11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
Rad sadrži elemente kao što su slike, broj strana i tabele tamo gde je to prirodno da se prikaže na taj način.
12. Da li su slike i tabele funkcionalne i adekvatne?
Slike i tabele su funkcionalne i adekvatne.

4.5 Ocenite sebe

Smatram da sam potpuno neupućen u oblast koju recenziram. Nikada se nisam bavio/učio Clojure kao ni bilo koji drugi funkcionalni jezik.

Glava 5

Dodatne izmene