

OPŠTI PODACI O PROJEKTNOM TIMU				
Br grupe	Broj indeksa	Ime	Prezime	Email adresa
5	PR81/2022	Dimitrije	Stanković	diki.zajecar@gmail.com
5	PR82/2022	Vojin	Jovanović	vojinsbb@gmail.com
Github link				
<a href="https://github.com/stankovic019/PRMuIS_projekat.git">https://github.com/stankovic019/PRMuIS_projekat.git</a>				

OPŠTI PODACI O PROJEKTU	
Naziv projekta:	Kviz "Kviskoteca"
TEHNIČKI OPIS PROJEKTA	
Sažetak:	Razvoj serverske aplikacije koja pruža klijentima mogućnost da odigraju tri igre kviza "Kviskoteca": <i>Anagrami</i> , <i>Igra pitanja i odgovora</i> i <i>Asocijacije</i> . Dva klijenta se mogu prijaviti da igraju na jednom serveru. Poeni se čuvaju za svakog igrača i na kraju igre se prikazuje pobjednik sa osvojenim brojem poena. Jedan klijent se može prijaviti tako da odigra trening igru, gde ne mora igrati sve četiri igre, već može da bira koje će igrati.
Primenjene metode:	<p><b>Multipleksiranje utičnica (socket multiplexing):</b> Centralni server (igra) istovremeno upravlja sa dva klijenta (igrača), obezbeđujući efikasnu komunikaciju.</p> <p><b>UDP i TCP komunikacija:</b></p> <ul style="list-style-type: none"> <li>• UDP se koristi za prijavu igrača (ako se prijavljuje samo jedan, šalje se i koje igre će igrati).</li> <li>• TCP se koristi slanje pitanja i odgovora tokom igre.</li> </ul> <p><b>Interakcija i obrada unosa za svaku različitu igru:</b> Zavisno od igre ponuditi adekvatnu interakciju (unos/izbor odgovora) i obradu unetih/izabranih vrednosti.</p> <p><b>Tok i stanje igre:</b> praćenje trenutno aktivne igre, poena koji se osvajaju i koliko koji igrač ima osvojenih poena. Na kraju igre, ispisati rezultate oba igrača.</p>
Opis projekta:	<p>Potrebno je razviti serversku aplikaciju koja upravlja igrom i klijentske aplikacije koje korisnicima pružaju da igra igre kviza "Kviskoteca". Aplikacije poseduju <i>sledeće karakteristike</i>:</p> <p><b>Igre:</b> Klijenti mogu igrati jednu od četiri igre:</p> <ul style="list-style-type: none"> <li>• <i>Anagrami</i>: Igračima se pošalje reč ili više reči a oni moraju poslati anagram (reč ili više reči koje imaju isto pojava svih slova)</li> <li>• <i>Igra pitanja i odgovora</i>: Na osnovu prikazanog pitanja i ponuđenih odgovora (Da/Ne), bira se tačan (unosom broja/slova ispred tačnog odgovora)</li> <li>• <i>Asocijacije</i>: Igrač može da zatraži da dobije informaciju o sadržaju polja u asocijaciji, unosom naziva polja ili da pokuša da pogodi sadržaj polja kolona ili konačnog rešenja.</li> </ul> <p><b>Server koji upravlja igrom:</b></p> <ul style="list-style-type: none"> <li>• <i>Reguliše prijavu igrača</i>: Ako se prijave dva igrača, priprema novu igru za obojicu. Ako se prijavi samo jedan, priprema igre samo za njega.</li> <li>• <i>Obrada odgovora i bodovanje</i>: na osnovu broja/tačnosti odgovora, dodeljuju se poeni svakom igraču.</li> </ul> <p><b>Računanje ukupnog broja poena:</b> osvojeni broj poena svakog igrača se prenosi u narednu igru.</p> <p><b>Prednost odgovaranja:</b> Omogućiti da se prvi poslati odgovor prvi uzima u obzir prilikom računanja poena, ako je korektan i da igrač koji ga je poslao dobije poene. Ako drugi igrač da korektan odgovor, dobija manje poena jer je bio sporiji. U igri asocijacije, više poena se dobija za davanje tačnih odgovora ako je igrač otkrio sadržaj manje polja.</p> <p><b>Primer upotrebe:</b> Na svaki pokrenuti server mogu se povezati do dva klijenta i igraju, određenim redom, jednu po jednu igru, dobijaju poene za korektne odgovore. Na kraju igrač sa više osvojenih poena pobeđuje.</p>

**Zadaci:**

1	<b>Zadatak:</b> Skica osnovne implementacije projekta	Rok: KT1	Broj poena: 2
	<b>Opis:</b> Napraviti blok-dijagram (šemu) koja prikazuje relacije između: <ul style="list-style-type: none"><li>• Kontrolora igre (servera), i</li><li>• jednog klijenta (igrača).</li></ul> Dijagram treba da uključuje tokove podataka i komunikacione protokole (UDP za prijavu, TCP za praćenje toka kviza).		
2	<b>Zadatak:</b> Implementacija osnovnih funkcionalnosti servera - određivanje detalja o igri	Rok: KT1	Broj poena: 3
	<b>Opis:</b> Na serveru se otvara UDP utičnica koja služi za prijavu igrača.  Klijent se prijavljuje slanjem poruke u formatu "PRIJAVA: [ime/nadimak], [lista igara koje igrač želi da igra, odvojene zarezima]", gde ta lista igara može da sadrži konstante "an", "po" i/ili "as". Ako je klijent pravilno poslao poruku o prijavi, server mu odgovara slanjem informacija o njegovoj TCP utičnici (IP adresa i broj porta). Kada se uspostavi TCP konekcija, server šalje klijentu (igraču) poruku "Dobrodošli u trening igru kviza Kviskoteka, današnji takmičar je [uneto ime/nadimak]".  <i>Napomena:</i> reč trening se ne ispisuje u pozdravnoj poruci, ako učestvuje više igrača (KT2). U treningu, igrač može najviše dva puta igrati igru "Pitanja i odgovori"		
3	<b>Zadatak:</b> Čuvanje podataka o igraču	Rok: KT1	Broj poena: 1
	<b>Opis:</b> Podaci o igraču se čuvaju u klasi <b>Igrač</b> .  Klasu <b>Igrac</b> čine polja: <ul style="list-style-type: none"><li>• Identifikacioni broj igrača (tipa <i>int</i>): dodeljuje se prilikom prijave igrača</li><li>• Ime/Nadimak igrača (tipa <i>string</i>)</li><li>• <i>Niz</i> vrednosti tipa <i>int</i>: broj poena po svakoj igri. Niz ima broj elemenata u skladu sa time koliko igara je igrač odlučio da igra.</li><li>• Uložen "Kvisko" (tipa <i>bool</i>): svaki igrač ima pravo da uloži "Kviska" pred određenu igru, čime mu se broj osvojenih poena duplira za tu igru (menja se na vrednost <i>true</i> pred samu igru, posle se vraća na <i>false</i>)</li></ul>		
4	<b>Zadatak:</b> Početak igre za klijenta	Rok: KT1	Broj poena: 1
	<b>Opis:</b> Kada se uspostavi konekcija sa TCP utičnicom i klijent dobije poruku o početku igre, treba da pošalje poruku START da bi započeo kviz.		
5	<b>Zadatak:</b> Igra "Anagrami"	Rok: KT1	Broj poena: 3
	<b>Opis:</b> Igra se implementira kao sopstvena klasa koja ima polja: <ul style="list-style-type: none"><li>• Reč ili više reči od kojih se pravi anagram (tipa <i>string</i>):</li><li>• Predložen anagram (tipa <i>string</i>)</li></ul> i metode: <ul style="list-style-type: none"><li>• Učitaj reč(i): Popunjava string koji predstavlja početnu reč/skup reči, od kojih se traži anagram. Raditi sa stringovima koji mogu da budu jedna reč (najviše 10 karaktera) ili da sadrže više reči. Stringove od kojih se pravi anagram definisati u klasi igre.</li></ul>		

	<ul style="list-style-type: none"> <li>• Proveri anagram: Proverava da li je predložen anagram sačinjen od istih slova kao i originalni tekst i da se svako slovo iz originalne reči (skupa reči) upotrebilo isti broj puta.</li> </ul> <p><i>Napomena:</i> Objekti ove klase se prave u okviru implementacije servera, a potrebne informacije za funkcionisanje igre se šalju klijentima. Klijent prosleđuje svoj anagram serveru, a server ga proverava unutar funkcije za proveru. Za tačan anagram, igrač dobija broj poena prema broju slova u reči (ukupan broj slova u više reči).</p>		
6	<b>Zadatak:</b> Igra "Pitanja i odgovora"	Rok: KT2	Broj poena: 3
<p>Opis: Igra se implementira kao sopstvena klasa koja ima polja:</p> <ul style="list-style-type: none"> <li>• Tekuće pitanje (tipa <i>string</i>): tekst pitanja koje se trenutno prikazuje</li> <li>• Tačan odgovor (tipa <i>bool</i>): tačno ili ne</li> <li>• Sva pitanja (Dictionary&lt;string,bool&gt;): predstavlja skup svih pitanja i njima uparene tačne odgovore.</li> </ul> <p>i metode:</p> <ul style="list-style-type: none"> <li>• Učitavanje pitanja: popunjava rečnik svih pitanja vrednostima - učitava se iz tekstualne datoteke (organizacija datoteke ostaje autorima).</li> <li>• Proveri tačan odgovor: Proverava da li je igrač poslao tačan odgovor na traženo pitanje.</li> </ul> <p><i>Primer:</i> Čovek ima tri bubrega. a) DA b) NE</p> <p>Klijent unosi i šalje slovo pored odgovara za koji smatra da je tačan</p> <p><i>Napomena:</i> Objekti ove klase se prave u okviru implementacije servera, a potrebne informacije za funkcionisanje igre se šalju klijentima. Klijent prosleđuje po jedan odgovor na svako od deset pitanja nazad serveru, gde se proverava tačan odgovor i klijent dobija poene. Za svaki tačan odgovor igrač dobija 4 poena. Omogućiti da se pitanja ne postavljaju uvek u istom redosledu i da nikad ne bude pitanje sa istim odgovorom postavljeno tri puta jedno za drugim.</p>			
7	<b>Zadatak:</b> Igra "Asocijacije"	Rok: KT2	Broj poena: 4
<p>Opis: Igra se implementira kao sopstvena klasa koja ima polja:</p> <ul style="list-style-type: none"> <li>• Niz nizova (tipa <i>string</i>): polja svake kolone i njeno konačno rešenje</li> <li>• Niz nizova (tipa <i>bool</i>): da li je otvoreno konkretno polje u svakoj od kolona</li> <li>• Konačno rešenje (tipa <i>string</i>)</li> </ul> <p>i metode:</p> <ul style="list-style-type: none"> <li>• Učitavanje asocijacije: vrednosti se definišu unutar klase. Napraviti više od dve asocijacije.</li> <li>• Otvaranje polja: Ispisuje se asocijacija tako da igrač vidi koja su otvorena polja.</li> </ul> <p><i>Primer:</i> A1: ??? A2: PSI A3: 3. Godina A4: Protokol A: ???</p> <p>Igrač unosi oznaku polja (da bi ga otvorio), ili poruku u formatu "[oznaka kolone]:[odgovor]" (da proba da pogodi rešenje kolone) ili "K:[odgovor]" za konačno rešenje</p> <p><i>Napomena:</i> Objekti ove klase se prave u okviru implementacije servera, a potrebne informacije za funkcionisanje igre se šalju klijentima. Klijent prosleđuje po jedan odgovor dok svaki klijent ne pošalje tri pogrešna odgovora na konačno rešenje, ili dok se ne reši kompletna asocijacija. Za svaku rešenu kolonu, korisnik dobija broj poena koji odgovara vrednosti koliko u toj koloni ima neotvorenih polja uvećano za 2. Za konačno rešenje se dobija 10 poena.</p>			
8	<b>Zadatak:</b> Istovremeni, neblokirajući rad sa više	Rok: KT2	Broj poena: 3

	klijenata		
	<b>Opis:</b> Omogućiti multipleksiranje nad TCP utičnicama koje server koristi za komunikaciju sa više igrača korišćenjem funkcije <i>Select</i> . Ažurirati postojeće rešenje, tako da radi sa više od jednog klijenta i da čuva informacije za svakog od njih. Napomena: Ispisivati broj poena nakon svake igre na serveru za svakog od igrača.		
9	<b>Zadatak:</b> Bodovanje i određivanje pobednika	Rok: KT2	Broj poena: 3
	<b>Opis:</b> Ako je određeni igrač uspeo da pogodi i prvi da pošalje odgovor dobija ukupan broj poena koji se daje za pitanje. Igrač koji kasni sa odgovorom (za svaki novopristigli odgovor, za svakih minut kašnjenja, igrač koji je poslao odgovor dobija 20% manje poena). Ovo važi samo za igru "Anagrami". Na kraju, potrebno je izračunati ukupan broj poena svakog igrača i ispisati konačnu tabelu poena na serveru, dok se klijentima šalje ko je pobednik/gubitnik. Ako dva igrača imaju isti broj poena, pobednik je onaj ko je ostvario osvojio više poena ulaganjem "Kviska".		
10	<b>Zadatak:</b> Skica ukupne implementacije projekta	Rok: KT2	Broj poena: 2
	<b>Opis:</b> Ažurirati blok-dijagram (šemu) dodavanjem: <ul style="list-style-type: none"> <li>• Više igrača koji simultano šalju odgovore.</li> <li>• Način na koji server obrađuje sve zahteve.</li> </ul>		