

OPŠTI PODACI O PROJEKTNOM TIMU				
Br grupe	Broj indeksa	Ime	Prezime	Email adresa
Stari studenti	PR69/2022	Djordje	Markovic	djmarkovic51@gmail.com
Stari studenti	PR12/2022	Andrija	Gradistanac	andra200313@gmail.com
Github link				
https://github.com/markovicdjordje/mrezeprojekat				

Zadaci:

1	Zadatak: Skica projekta	Broj poena: 2
	<p>Opis: Napraviti blok dijagram koji ilustruje osnovnu arhitekturu sistema:</p> <ul style="list-style-type: none">• Centralni server za koordinaciju taksi vozila i klijenata.• Jedno taksi vozilo koje komunicira sa serverom putem TCP protokola.• Jeden klijent koji šalje zahtev serveru putem UDP protokola. <p>Dijagram treba da obuhvati:</p> <ul style="list-style-type: none">• Prijem zahteva od klijenta putem UDP-a.• Slanje zadatka vozilu putem TCP-a.• Povratnu informaciju o statusu zadatka klijentu. <p>Proširiti dijagram projekta tako da obuhvati:</p> <ul style="list-style-type: none">• Više taksi vozila povezanih sa serverom putem TCP protokola.• Više klijenata koji šalju zahteve serveru putem UDP protokola.• Dinamičko upravljanje zadacima servera u skladu sa trenutnim statusom vozila i zahtevima klijenata.	
2	Zadatak: Inicijalizacija servera i osnovna komunikacija sa taksi vozilom i klijentom	Broj poena: 2
	<p>Implementirati centralni server koji:</p> <ul style="list-style-type: none">• Pokreće TCP soket za povezivanje vozila.• Pokreće UDP soket za prijem zahteva od klijenata.• Prihvata osnovne podatke od klijenta (početna i krajnja tačka putovanja).• Prihvata osnovne podatke od vozila (trenutna pozicija i status). <p>Testirati prijem jednog zahteva od klijenta i odgovarajuće slanje zadatka vozilu.</p>	
3	Zadatak: Definicija klasa za taksi vozilo, klijenta i zadatak	Broj poena: 2
	<p>Opis: Implementirati sledeće klase:</p> <ul style="list-style-type: none">• TaksiVozilo: Trenutne koordinate, status (slobodno, odlazak na lokaciju, vožnja), pređena kilometraža, zarada.• Klijent: Početna i krajnja tačka, status zahteva (čekanje, prihvaćeno, završeno).• Zadatak: ID klijenta, ID vozila, status zadatka (aktivan, završen), pređena razdaljina. <p>Klase moraju podržavati serijalizaciju i deserializaciju za prenos podataka putem mreže pomoću MemoryStream-a.</p>	
4	Zadatak: Osnovna simulacija dodavanja i izvršavanja zadatka	Broj poena: 2

	<p>Opis: Implementirati osnovnu simulaciju dodavanja zadatka:</p> <ul style="list-style-type: none"> • Klijent šalje zahtev za prevoz serveru putem UDP-a. • Server pronađe najbliže slobodno vozilo na osnovu koordinata klijenta. • Server šalje zadatak vozilu putem TCP-a, uključujući podatke o klijentu. • Vozilo prelazi u stanje "vožnja" i simulira prevoz klijenta do krajnje tačke. • Server ažurira status vozila i zadatka nakon završetka vožnje. 	
5	Zadatak: Upravljanje zadacima i raspodela zahteva klijentima i vozilima	Broj poena: 3
	<p>Opis: Implementirati logiku servera za upravljanje zadacima:</p> <ul style="list-style-type: none"> • Server vrši obradu poruka od klijenata i vozila pomoću multipleksiranja. • Raspodela zadataka najbližem slobodnom vozilu na osnovu koordinata klijenta. • Slanje odgovarajuće poruke klijentu o prihvatanju zahteva i približnom vremenu dolaska vozila. • Dinamičko ažuriranje statusa vozila i zadatka u toku vožnje. • Obrada završetka vožnje i ažuriranje baze podataka servera. <p>Implementirati vizuelizaciju sistema na serveru:</p> <ul style="list-style-type: none"> • Tabelarni prikaz trenutnog statusa svih vozila (pozicija, status, zarada, kilometraža). • Tabelarni prikaz svih aktivnih zahteva klijenata (početna i krajnja tačka, status). • Dinamičko osvežavanje prikaza u svakom ciklusu sistema. 	
6	Zadatak: Simulacija vožnje sa praćenjem pozicija vozila	Broj poena: 2
	<p>Opis: Implementirati simulaciju kretanja vozila:</p> <ul style="list-style-type: none"> • Vozilo simulira kretanje prema klijentovoj početnoj tački i zatim prema krajnjoj tački. • Server periodično ažurira pozicije vozila i šalje ažurirane podatke klijentu. • Uključiti algoritam za izračunavanje razdaljine i vremena potrebnog za dolazak vozila na odredište. 	
7	Zadatak: Logika završetka zadatka i ažuriranje performansi vozila	Broj poena: 2
	<p>Opis: Implementirati završetak zadatka:</p> <ul style="list-style-type: none"> • Po završetku vožnje, vozilo obaveštava server o naplaćenom iznosu, pređenoj razdaljini i završenom zadatku. • Server ažurira klasu vozila (ukupan broj mušterija, zarada, kilometraža). • Prikazati ažurirane podatke na terminalu servera u vidu liste. 	