

Korisnički zahtjevi, G3


Alat za praćenje osobnih financija

Sažetak

Ovaj dokument definira zadaću za studente grupe G3 na kolegiju Programsko inženjerstvo (22/23). Dokument predstavlja osnovu za izradu studentskog projekta. Svaka sumnja na plagijat rezultirat će detaljnom analizom kôda rješenja i usmenim ispitivanjem.

Transkript razgovora s potencijalnim poslovnim partnerom

Jesen je 2026. Svježe ste diplomirali. Otišli ste na kavu s vašom prijateljicom s ekonomskog smjera koju ste upoznali tijekom studiranja. Ona vas je pozvala na kavu kako bi vas angažirala za rad na programskom proizvodu koji joj treba. Vaša uloga u razgovoru jest „programski inženjer“. Slijedi transkript vašega razgovora.

	Napomena
Situacija izložena u dokumentu u potpunosti je izmišljena.	
Likovi ne postoje i nisu bazirani na stvarnim osobama.	
Svaka sličnost sa stvarnim osobama je nenamjerna.	

Detaljno analizirajte razgovor i izvucite natuknice o svim bitnim funkcionalnostima koje uočite.

POČETAK TRANSKRIPTA

Programski inženjer: „Hej, bok! Kako si?“

Bivša kolegica: „Bok! Ma super. Hvala na dolasku. Čuj, mislim da si ti prava osoba za ovaj posao. Ti znaš programirati, zar ne? Onako, ozbiljno? Imaš iskustva s time?“

Programski inženjer: „Pa naravno, bavim se time još od kolokviranja programskog inženjerstva na drugoj godini! Mojoj je grupi asistent dao baš neki ozbiljan zadatak, a pohvalio je moje rješenje.“

Bivša kolegica: „E super! Gle, ja ti sad radim u jednoj *cool* tvrtki, imamo sve digitalno u ERP sustavu ZAP. To ti je ono, tvrtka dobije račun, imaš datum dospijeca, proučiš, proknjižiš, imaš razne detaljne stavke analitike, sve je u sustavu. Tvrtka dobije dostavnicu, proknjižiš, zaprimiš, ažurira se stanje – “

Programski inženjer: „Em, oprost, ali nadam se da me nisi zbog toga zvala. To ti je meni strašno dosadno, a zbog riječi 'proknjižiti' mi je i danas muka.“

Bivša kolegica: „Ma ne, ne brini, ono što te trebam uopće nema veze s tim sustavom. Poanta je da mi nedostaje takav sustav doma. Nešto što mi može pratiti moja trošenja. Ove cijene danas, i inflacija... Ulje 2€. Zbog prelaska na eure ne mogu procijeniti trošim li previše ili premalo. Meni samo treba sustav koji će mi reći koliko trošim na što i trebam li smanjiti potrošnju.“

Programski inženjer: „U redu, razumijem. Kako zamišljaš takav sustav?“

Bivša kolegica: „Pa vidiš, moja neka ideja je da unesem trošak, npr. račun iz trgovine 8€, uz primjerice komentar: 'Ulje, mlijeko, jaja'. I onda negdje imam pregled svih tih troškova. Svaki trošak ima svoj opis, postoji pregled svih troškova, pregled troškova samo za neki vremenski period, i tako.“

Programski inženjer: „Zar već ne postoji aplikacija za to?“

Bivša kolegica: „Ma imam sve te moderne *FinTech* aplikacije na mobitelu, ali sve toliko bespotrebno kompliciraju! Meni treba jednostavan i brz pregled na računalu. Ne volim obrađivati podatke na ekranu mobitela. Trenutno u bilješkama na mobitelu držim kategorizirane sve troškove, ali sada tih bilješki ima preko 200. Čovjek se više ne može snaći u tome. Želim jednostavnu aplikaciju na laptopu koja mi daje pregled mojih troškova kategoriziranih po vrsti.“

Programski inženjer: „Kako misliš, po vrsti?“

Bivša kolegica: „E vidiš, jako mi je bitno da se troškovi dijele na vrste! U bilješkama sve troškove označavam brojem i iznimno mi je bitno da ti isti brojevi budu i negdje u aplikaciji uz svaku vrstu. Tako sam se jednostavno naviknula. Popisala sam ti vrste troškova na papir.“

U tom trenutku iz džepa vadi te vama predaje idući papirić.

Vrste mojih troškova:

- hrana i piće - 1
 - > pekara - 11
 - > ostalo - 12
 - kupovine - 2
 - > namirnice - 21
 - > svakodnevni život - 22
 - > velike kupovine - 23
 - dugovanja i potraživanja - 3
 - > nekome posudim novce - 31
 - > netko mi uplati novce - 32
 - mjesečna članstva - 5
 - > npr. boks, teretana itd.
 - nezgode - 6
 - > u kući - 61
 - > vani - 62
- džeparac - 33
bonus na plaću - 35
najamnina - 36

Programski inženjer: „Ideš na boks?“

Bivša kolegica: „Da, krenula sam prošli mjesec. To mi je prva takva aktivnost od kad držim evidenciju, pa ju u mobitelu bilježim brojem 51. Općenito, što se tih mjesečnih članarina tiče, planiram se nakon mnogo vremena vratiti u teretanu. Navodno se uskoro moderniziraju nekim sustavom za praćenje članova. Nego, kada odlučim ići, jednostavno ću bilježiti teretanu brojem 52. Nakon toga, ako odlučim ići na ples, onda će to postati broj 53. Čak i ako prestanem ići u teretanu, ples bi u tom slučaju ostao broj 53 – te šifre ne smiju mi se mijenjati jer u glavi sve pamtim po brojevima. Dakle, svaka nova aktivnost je za jedan broj više od prethodne, bez obzira na ostale. Kužiš?“

Programski inženjer: „U redu, ali zašto nema stavke s brojem 4?“

Bivša kolegica: „Aha, to. Ma mrzim taj broj. Ne podnosim ga. I da, taj broj ne smije biti niti u jednoj šifri, nigdje i nikako. Je li to teško implementirati? To bi mi bilo strašno važno.“

Programski inženjer: „Čudna odluka, ali nije problem. Kako god, što je s ostalim stavkama?“

Bivša kolegica: „Sve kako je i napisano. Evo, npr., ovu kavu ću bilježiti pod šifrom 12. Te ostale brojke su fiksne i ne bih željela da se mijenjaju. Ponekad imam želju previše komplicirati pa želim da me program spriječi u tome da stvaram beskonačno mnogo šifri za svaku sitnicu koju kupujem. Jučer navečer definirala sam ovaj popis kao finalni i želim da bude fiksni. Jedino gdje se smiju dodavati i brisati šifre stavki su te mjesečne članarine. I još jedna stvar. Iznimno mi je važno da mogu učitati dosadašnjih 200 bilješki u taj program. Bilo bi predobro da se podaci mogu učitati iz CSV datoteke. To mi nije problem sastaviti, a uštedjelo bi mi gro vremena. U tom slučaju, ti samo kažeš kako da složim stupce. Vrijednosti po stupcima mogu biti šifra, iznos potrošenog novca, komentar uz transakciju. Automatsko učitavanje preko CSV datoteke bi mi moglo uštedjeti mnogo vremena i truda. Ako ti je to previše posla, nema veze, ali onda barem da mogu ručno unositi te stavke.“

Programski inženjer: „Razumijem, vidjet ću što ću s time. Još nešto?“

Bivša kolegica: „I još jedna stvar. Želim imati mogućnost da proknjižim – ups, upišem svoju mjesečnu plaću, a da se onda prikaže neka statistika koja mi vremenski označava kako moja količina novaca s početka mjeseca pada. Možda neko predviđanje s obzirom na prosječnu dnevnu potrošnju koliko ću imati na računu do kraja mjeseca i koliko još dana imam do nule na računu s tim tempom trošenja. Ako bi ta statistika mogla biti neki graf, još bolje. U suprotnom, bit će super i sa samo tekstualnim informacijama.“

Programski inženjer: „Vidjet ću još to s obzirom na vrijeme. A koliko ti je bitna sigurnost?“

Bivša kolegica: „Pa, svaki drugi vikend idem doma i onda mlađi buraz igra igrice na mom laptopu, iako mu ne dam. Želim neki sustav prijave čisto da on ne pokrene program i krene neke gluposti kucati unutra u moju evidenciju. Ali taj sustav prijave mora se moći uključiti po potrebi, jer ne želim da me program gnjavi prijavom dok njega nema. Inače, već mi je jednom i instalirao neki virus na laptopu, pa sam izgubila sve podatke. Ne znam možeš li nekako osigurati podatke da ne ovise o laptopu?“

Programski inženjer: „Osmislit ću neku vrstu jednostavne prijave kako nitko ne bi mogao tek tako upasti. A što se podataka tiče, to ćemo riješiti vanjskom bazom podataka.“

Bivša kolegica: „Super. Što ćemo sada?“

Programski inženjer: „Analizom snimke ovog razgovora definirat ću zahtjeve. Nakon toga dizajnirat ću sustav, a potom kreće programiranje. Sva sreća da snimam razgovor diktafonom ispod stola.“

Bivša kolegica: „Čekaj?! Snimaš naš razgovor?!“

KRAJ TRANSKRIPTA

(bitne informacije na idućoj stranici)





Napomena

U stvarnome svijetu, uloga „programski inženjer“ bila bi u biti „arhitekt sustava“ (engl. *System Architect*) ili „vlasnik proizvoda“ (engl. *Product Owner, PO*).

OBVEZNI ELEMENTI SPECIFIKACIJE

Specifikacija treba sadržavati minimalno:

- 8 funkcionalnih zahtjeva, od toga minimalno:
 - 1 zahtjev u domeni prikaza dohvaćenih podataka
 - 1 zahtjev u domeni unosa podataka
 - 1 zahtjev u domeni izrade izvještaja
 - 1 zahtjev u domeni prikaza statističkih podataka
- 2 nefunkcionalna zahtjeva, od toga minimalno:
 - 1 zahtjev u domeni izgleda softvera
 - 1 zahtjev u bilo kojoj drugoj domeni