



А К А Д Е М И Ј А  
ТЕХНИЧКО-ВАСПИТАЧКИХ  
СТРУКОВНИХ СТУДИЈА



## PROGRAMSKI ALATI ZA RAZVOJ SOFTVERA

### Drugi kolokvijum

#### Zadatak 1: Sistem za upravljanje informacijama o kućnim ljubimcima

Napraviti prototip aplikacije za upravljanje informacijama o kućnim ljubimcima koja omogućuje:

1. Dodavanje novih ljubimaca u sistem.
2. Kreiranje različitih vrsta ljubimaca sa specifičnim karakteristikama.
3. Centralizovano čuvanje i upravljanje informacijama o svim ljubimcima.
4. Praćenje i obaveštavanje vlasnika o promenama vezanim za ljubimce, kao što su promene u zdravstvenom stanju ili vlasništvu.

Funkcionalnosti:

1. Sistem mora da podrži dodavanje različitih vrsta ljubimaca, kao što su psi, mačke, ptice, itd.
2. Informacije o ljubimcima treba da uključuju ime, vrstu, rasu, starost i dodatne karakteristike specifične za vrstu ljubimca. Možete staviti i kontakt vlasnika (ime, telefon, ili mejl).
3. Sistem treba da obezbedi centralni registar ljubimaca koji podržava operacije pretrage i izmene informacija.
4. Vlasnici treba da budu obavešteni o promenama vezanim za ljubimca, kao što su promena zdravstvenog stanja ili približavanje termina za vakcinaciju.
5. Omogućiti i dodatne funkcionalnosti kao što su brisanje ljubimca iz sistema ili filtriranje ljubimaca prema različitim kriterijumima (vrsta, starost, itd, dovoljan je jedan kriterijum).

Glavni zahtevi:

1. Opišite sva tri dizajn paterna koji se ovde koriste, koji su oni, zašto su oni bitni, i kakvu ulogu generalno imaju u ovom sistemu. Budite što više detaljni i slobodno navedite još neku njihovu primenu. Smatrajte ovo teorijskim delom kolokvijuma.
2. Nacrtajte UML Class dijagram i Use Case dijagram ako želite (on nosi 3 dodatna poena).
3. Izlistajte sve klase i interfejse koji su vam ovde potrebni i naglasite koji patern pokrivaju.
4. Napišite bar 5 Junit ili py testova (sami birate jezik) gde ćete pokriti ispravne unose, ali i one sa greškom, kao i null i prazne.
5. Opišite algoritam koji koristite za sortiranje ili filtriranje ljubimaca i u kojoj strukturi (kolekciji) podataka ste planirali da čuvate informacije o ljubimcima (ovaj deo nosi 2 dodatna poena).

Primer korisničkog scenarija: Vlasnik Aca unosi novog psa u sistem sa podacima: ime "Max", rasa: Pitbull, starost 3 godine. Nakon godinu dana, dolazi vreme za novu vakcinaciju. Sistem automatski obaveštava vlasnika o ovom događaju. Vlasnik takođe može da pregleda sve ljubimce koje ima.

U folderu sa svojim radom, otvorite podfolder “**1 - Veterinar**” i tu ubacite sve fajlove koje ste kreirali, uključujući crteže, kod (pseudokod), tekstualne objašnjenja i primere za testiranje.

## Zadatak 2: Aplikacija za naručivanje kafe

U okviru ovog zadatka neophodno je kreirati prototip aplikacije za naručivanje kafe. Ključni elementi su dati u nastavku:

1. Naslov aplikacije, *Akademaska Kafica*, kao i prateći logo (sličica kvadratnog oblika).
2. Dugmeta za **Novu porudžbinu, Obradu trenutne porudžbine i Reset unosa**.
3. Tekstualno polje sa labelom ili placeholderom za ime korisnika koje treba napisati na čaši (*Starbucks* fazon).
4. Padajuća lista sa tipom kafe koju korisnik može naručiti (Espresso, Kupućino, Makijato...oko pet vrsta kafe je poželjno). Cene kafa prilagodite da budu između 150 i 200 dinara.
5. +/- polje sa labelom za dodavanje šećera, može se dodati maksimalno 5 kašičica šećera. Šećer je besplatan.
6. Adekvatna dugmeta za odabir blenda, korisnik može odabrati *Arabica 100%* ili *Balkan Blend*.
7. Adekvatna dugmeta za odabir dodataka, korisnik može odabrati mleko, šlag, rum, cimet i dodatni espresso šot. Dakle, može odabrati jedan ili više dodataka, a svakako može naručiti kafu i bez toga. Cena dodataka je 50-100 dinara, odredite sami.

Ove elemente treba pažljivo skicirati preko odgovarajućeg softvera i objasniti kako se definišu u Javi ili Pythonu, sami birate jezik. Neophodno je svaki detaljno objasniti i napisati komandu za kreiranje. Sliku interfejsa aplikacije predajete zajedno sa tekstualnim opisima. Potrebno je i iskucati kod (ili pseudokod) za metodu koja će pokupiti korisnikove unose, izračunati ukupnu cenu poručene kafe i snimiti podatke o porudžbini u TXT, XML ili JSON fajlu. Takođe je potrebno ispisati i bar 5 metoda za testiranje, gde ćete pritom proveriti:

- Null vrednosti
- Prazne tekstualne vrednosti
- True/False vrednosti
- Ispravne numeričke unose
- Neispravne numeričke unose

U folderu sa svojim radom, otvorite podfolder “**2 - Akademaska Kafica**” i tu ubacite sve fajlove koje ste kreirali, uključujući crteže, kod (pseudokod), tekstualna objašnjenja i primere za testiranje.

---

### Bodovanje:

Prvi zadatak (20 + 5 poena):

tekstualna objašnjenja 10 poena, testovi 5 poena, grafikoni 5 poena, moguć i bonus od 5 poena.

Drugi zadatak (20 poena)

skica 5 poena, testovi 5 poena, metoda za izračunavanje 5 poena, tekstualna objašnjenja 5 poena.