

SVEUČILIŠTE U ZADRU  
Preddiplomski stručni studij informacijskih tehnologija

**Simulacija Kladenja**  
**Program-igrice**

Učenik: Šime Rončević

Nastavnik: Doc. dr. sc. Ante Panjkota

Predmet: Osnove objektnog programiranja

Zadar, 2022.

## Problem:

Problematika ovog zadatka je napraviti simulaciju nekog nogometnog turnira u kojem će korisniku biti moguće izvesti interakciju u obliku klađenja na pobjednika. Turnir ima osam klubova i tri kola. Trebalo je napraviti prozor koji će predstavljati glavnu igru. U tom prozoru korisnik će birati favorita i ostavljati određeni ulog novca (Valuta nije definirana). Svaki klub će imati naizmjenično danu kvotu, te će se ta kvota množiti s ulogom. Korisnik također mora imati mogućnost da odustane od igre i pokupi svoj novac. Da bi korisnik mogao vidjeti stanje pojedinog kola mora imati mogućnost otvaranja prozora koji će mu prikazati stanje turnira u pojedinom kolu. Korisnici su definirani kao objekti i spremaju se u poseban bin file kojeg ćemo kasnije iščitati za prikaz korisnika koji su igrali igru.

Dodatni zadatak se naziva „Bonus igra“ koju korisnik može, ali i ne mora zaigrati. Bitno je da kad jednom počne igrati da više ne može odigrati bonus igru. Bonus igra će tražiti od korisnika da odabere svoga favorita te će dati naravno svoj ulog. Nakon što zaigra bonus igru prozor se zatvara, a korisnikovo predviđanje i mogući dobitak se pamte do kraja igre. Ako je korisnik pogodio dobiva dodatnu nagradu. Zbog jednostavnosti odlučio sam da jednom kada korisnik ostane bez novca više ne može igrati bez obzira na dodatnu igru. Najveći problem ovog programa je taj što se igra konstantno mijenja. Sve vrijednosti, klubovi i rezultati se mijenjaju na skoro svaki klik mišem. To je samo po sebi jako podložno raznim greškama i „exceptionima“.

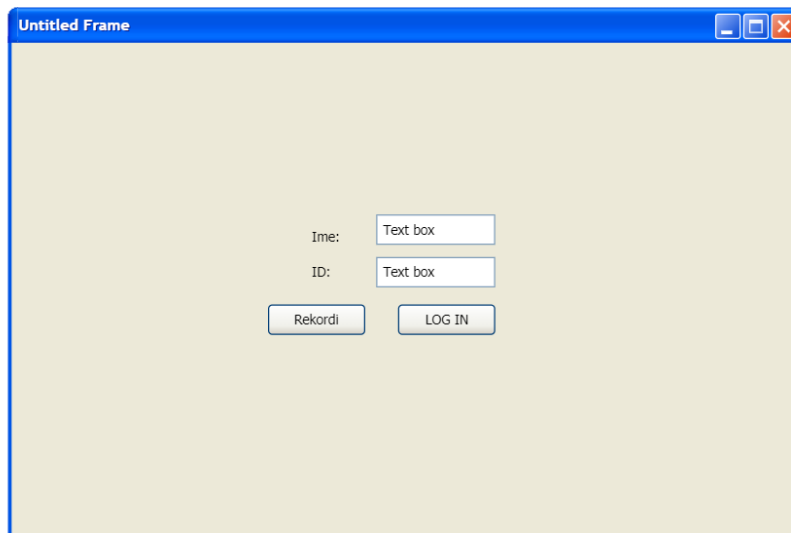
## Rješenje:

### 1. Korak

Ovom zadatku moguće je pristupiti na različite načine, ali možda najbolji je da se prvo naprave sva grafička sučelja pomoću Java Swing. Samo po sebi sučelje je moglo biti neuredno (u smislu „zgužvano“). Ja sam se odlučio na micanje elemenata jedne od drugih pomoću različitih praznih „JLabela“. Tako je kod malo duži, ali vizualno grafičko sučelje je puno ljepše. Nakon što smo vizualno kreirali naš program, potrebno mu je bilo dodati dušu, odnosno program koji će se vrtjeti u pozadini i omogućavati korisniku da igra igrice.

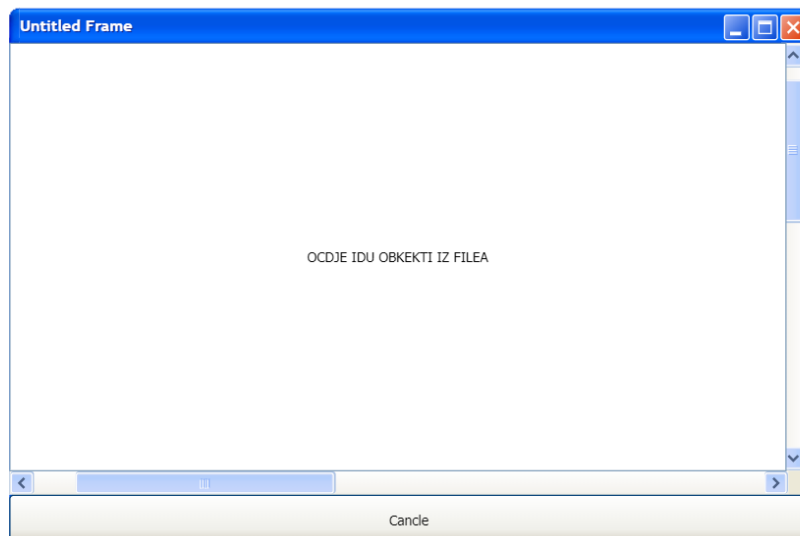
Početna stranica je prozor koji se otvara na početku pokretanja programa. Vizualno je jednostavna te pokreće Glavnu igru ili prikaz korisničkih računa koji su igrali i osvajali nagrade. Turnir se sastoji od jedne klase, te je možda najjednostavniji dio programa. Bonus igra se također sastoji samo od jedne klase, dok sam klasu „GlavnaIgra“ napravio pomoću dvije dodatne „GlavnaIgra\_UPP“ i „GlavnaIgraDown“. Glavna igra down je zapravo sučelje naše igre čiji je osnovni zadatak da vizualno odvoji odabir timova od drugih postavki. Glavna igra Down posjeduje više varijabla i ona uzrokuje pokretanje više različitih metoda, te otvara prozor za bonus igru i za prikaz turnira.

Početna stranica

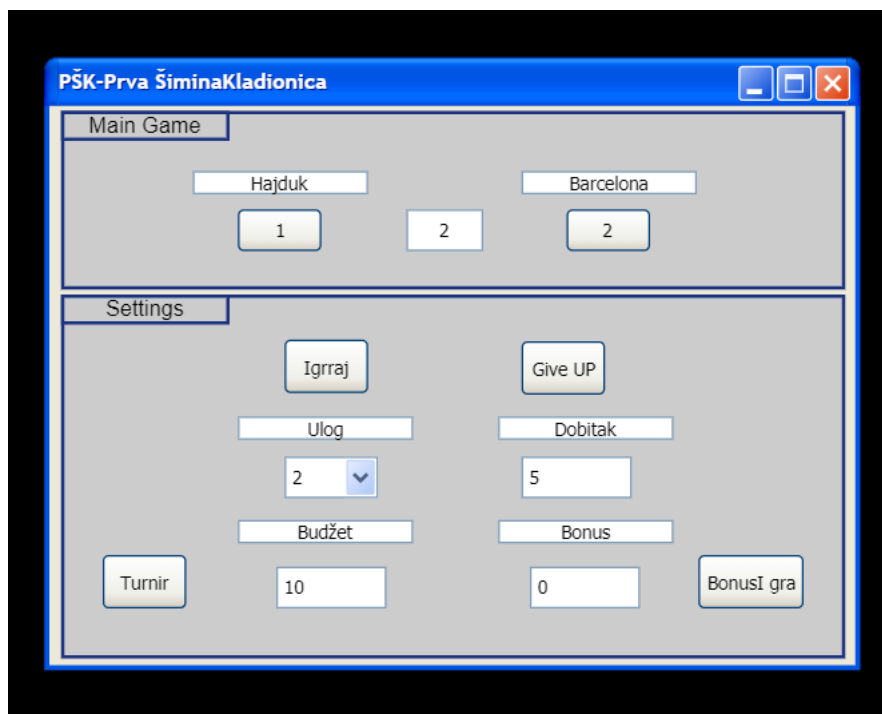


The screenshot shows a web browser window titled "Untitled Frame". The page has a light beige background. In the center, there are two text input fields. The first is labeled "Ime:" and the second is labeled "ID:". Below these fields are two buttons: "Rekordi" on the left and "LOG IN" on the right.

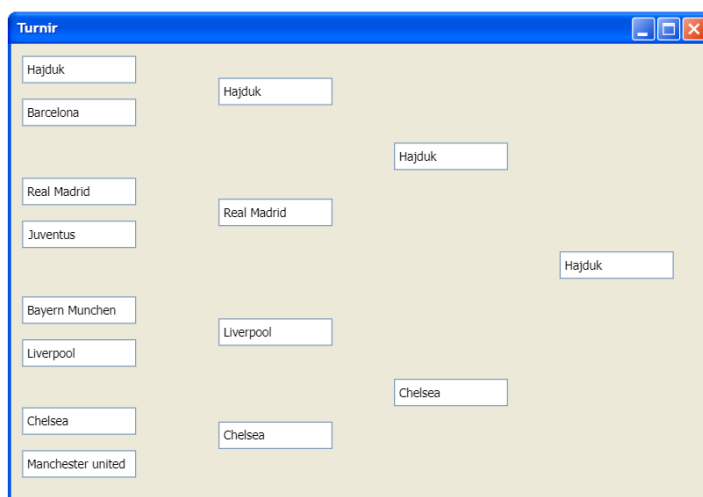
Ako kliknemo na botun „Rekordi“ dobivamo sljedeće:



Pokretanjem botuna „LOG IN“ dobivamo sljedeći prozor



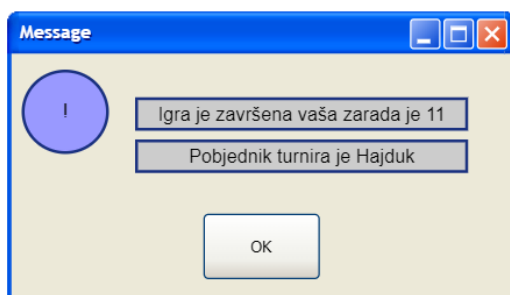
Klikom na botun turnir otvara se



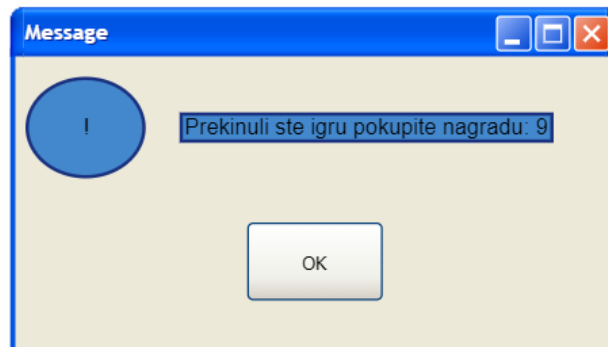
Kada kliknemo na botun Bonus Igra

The screenshot shows a window titled "Bonus igra" with a blue title bar and standard Windows window controls. The main area has a light beige background. At the top, there is a label "Predvidite Pobjednika" above a dropdown menu currently showing "Hajduk". Below this, there are two input fields: "Kvota" with the value "4" and "Ulog" with a dropdown menu showing "1". To the right of the "Ulog" dropdown is another input field with the value "4", followed by a label "mogući dobitak". At the bottom, there are two buttons: "Submit" on the left and "Candle" on the right.

Kada igra završi dobivamo sljedeće



Ukoliko smo prekinuli igru dobit ćemo sljedeće:



## 2.Korak

Zbog jednostavnosti počeo sam od bonus igre s tim da sam u Glavnoj igri napravio određene varijable koje su mi bile potrebne. Osnovni problem Bonus igre je bio taj što treba mijenjati kvotu ovisno o odabranom timu, Te istovremeno mijenjanje mogućeg dobitka koji je umnožak naizmjenično generirane kvote i uloga. To znači da ista pravila o ažuriranju vrijednosti vrijedi i za mijenjanje uloga. Ulog i Odabir tima sam riješio u obliku „JComboBox-a“ koji smanjuje mogućnost grešaka poput krivog unosa. U pozadini je najvažnija metoda za računanje naših vrijednosti. Ona također onemogućuje korisniku da se kladi više od iznosa kojeg ima tako da će blokirati „submit“. Korisnik ne može ući u interakciju s niti jednim „JTextFieldom“ jer naš program iščitava vrijednosti iz njih. Nakon što korisnik klikne na „Submit“ potencijalni dobitak i predviđanje pobjednika.

## 3.Korak

Nakon što sam napravio bonus igru odlučio sam raditi glavnu igru bez koje klasa turnir ne može funkcionirati. Glavna igra predstavljala je najveći problem, pa tako ima i najveći broj metoda. Glavna igra uz elemente sučelja također posjeduje nepromjenjive nizove koji sadrže sve moguće kvote i sve moguće klubove. Radi jednostavnosti stavljeno je da je unaprijed određen turnir. Postoji niz nasumično pomiješanih klubova koji će se kasnije u drugim kolima ažurirati, zatim „HashMap“ koji čuva naizmjenično određene klubove s njihovim naizmjenično određenim kvotama. Također imamo varijablu za čuvanje pobjednika iz bonus igre i još jedno koja će prihvatiti pravog pobjednika. Metoda izračunaj dobitak će nam kad god je zovnemo izračunati dobitak na temelju danih vrijednosti. Koristimo je u klasi „GlavnaIgra\_UPP“ i „GlavnaIgraDown“. Metoda pokreni igru u pozadini je prvi korak u pokretanju naše igre. Ona će odrediti

sve početne vrijednosti. Sastoji se od dodatnih metoda od kojih svaka odrađuje svoj dio posla. Metoda „NovaUtak“ je bila najkompliciraniji dio ovog zadatka jer je upravo ona ta koja prebacuje utakmice i omogućuje pamćenje rezultata. Sastoji se od dvije dodatne metode te ona ukupno određuje pobjednika, pregledava da li smo pogodili i na temelju toga nam pripisuje ili oduzima novac s računa. Ona je također zaslužna za ažuriranje naše liste koja posjeduje pomiješane klubove. Svaki put kad se ova metoda pokrene naša će lista biti sve manja.

„GlavnaIgra\_UPP“ nam omogućuje da korisnik mijenja svoju odluku o izboru te na račun toga konstantno ažurira podatke koristeći određenu metodu.

„GlavnaIgraDown“ također ažurira podatke odnosno dobitke i nama puno važnije ona pokreće novu fazu igre ako kliknemo na „IGRAJ“. Sve greške su spriječene tako što se botun „IGRAJ“ deaktivira ako vrijednost nije određena.

## **4.Korak**

Nakon što sam ispitao dosad napravljeni dio programa koji može raditi neovisno o klasi turnir, mogao sam se prebaciti na rješavanje tog problema. Zbog jednostavnosti odredio sam da se turnir ažurira tek nakon odigranog kola. Dodatne liste U klasi „GlavnaIgra“ će nam pomoći da sačuvamo vrijednost pojedinog kola. Oni će pak svaki put pri pokretanju našeg turnira biti prebačene u klasu Turnir. Tu će se zadovoljavati određeni uvjeti. Ako klasa svakog kola ima odgovarajući broj članova odnosno klubova naši TextFieldovi će se ispuniti. Samo rješenje ne zvuči komplicirano, ali način na koji je bilo potrebno riješiti taj problem zahtjeva određeni fokus i strpljenje. Na kraju za pobjednika se uzima podatak iz „GlavnaIgra“.

## **5.Korak**

Nakon što sam napravio cijelu glavnu igru i ustvrdio da radi mogao sam dodati Početnu stranicu koju ću koristiti za stvaranje korisnika, odnosno igrača i koja će pokrenuti našu igru ili nam prikazati korisnike koji su od prije igrali igru. Korisnici su objekti koji su realizirani pomoću jednostavne klase, dok njihovo spremanje u file i iščitavanje iz filea radi zasebna klasa. Zbog jednostavnosti programa naša početna strana može pokrenuti našu igru samo jednom.

```

classDiagram
    class GlavnaIgra {
        +ArrayList<String> kolo
        +String pobjednik
        +ArrayList<String> kolo2
        +ArrayList<String> kolo3
        +int trenutnoStanjeNaRacunu
        +void initComponents()
        +void pokrenilgruUPozadini()
        +void layoutAll()
        +void prijelazIgre()
        +void novaUtak()
        +void pomjesajKlubove()
        +void getSveKvote()
        +void provjeriZaGasiti()
        +void izracunajDobitak()
        +void jesamLiPogodio()
        +void jesmoLiDobiliBonus(String)
        +ArrayList<String> kolo
        +ArrayList<String> kolo2
        +String pobjednik
        +int trenutnoStanjeNaRacunu
        +ArrayList<String> kolo3
    }
    class GlavnaIgra_UP {
        +GlavnaIgra_UP()
        +JTextField ime2Kluba
        +JTextField ime1Kluba
        +JTextField ispisKvote
        +void layoutComponents()
        +void activateGlavnaUPP()
        +void ininitComponents()
        +void setBorders()
        +JTextField ime2Kluba
        +JTextField ime1Kluba
        +JTextField ispisKvote
    }
    class PocetnaStr {
        +PocetnaStr()
        +ArrayList<Korisnik> korisniciZF
        +Korisnik noviKorisnik
        +GlavnaIgra Gt
        +void initComponents()
        +void activatePocetnaStr()
        +void layoutComponents()
        +Korisnik noviKorisnik
        +GlavnaIgra Gt
        +ArrayList<Korisnik> korisniciZF
    }
    class GlavnaIgraDown {
        +GlavnaIgraDown()
        +BonusIgra Bt
        +void layoutComponents()
        +void setBorders()
        +void activateGllgruDown()
        +void ininitComponents()
        +BonusIgra Bt
    }
    class BonusIgra {
        +BonusIgra()
        +void initComponents()
        +void layoutComponents()
        +void generirajKvote()
        +void izrDobitak()
        +void activateBonusGame()
    }
    class Rekordi {
        +Rekordi()
        +void layoutAll()
        +void initComponents()
        +void butnCancel()
        +void unesiTekst()
    }
    class Main {
        +Main()
        +PocetnaStr pocetnaStr
        +void main(String[])
        +PocetnaStr pocetnaStr
    }
    class KorisniciZF {
        +Korisnik(String, String)
        +int novacNaRacunu
        +String toString()
        +int novacNaRacunu
    }
    class CitanjeFilea {
        +CitanjeFilea()
        +boolean fileExists(String)
        +ArrayList<Korisnik> loadKorisnici()
        +void saveKorisnici(ArrayList<Korisnik>)
    }
    class Turnament {
        +Turnament()
        +void initComponents()
        +void layoutComponents()
        +void ispisiTurnir()
    }

    GlavnaIgra --> GlavnaIgra_UP : GU
    GlavnaIgra --> PocetnaStr : create
    GlavnaIgra --> GlavnaIgraDown : GU
    GlavnaIgraDown --> BonusIgra : create
    BonusIgra --> Rekordi : GU
    Rekordi --> Main : create
    Main --> PocetnaStr : create
    Main --> KorisniciZF : create
    Main --> CitanjeFilea : create
    Main --> Turnament : create
    
```

7