

# Projekat:Connect 4

Igra Connect 4 predstavlja klasičnu igru strategije u kojoj dva igrača ili jedan igrač protiv AI naizmjenično postavljaju figure u tabli sa 7 kolona i 6 redova. Cilj je postaviti četiri diska iste boje u vertikalnu, horizontalnu ili dijagonalnu liniju, pri čemu pobjeđuje igrač koji prvi postigne ovaj uslov.

## G.0 ČLANOVI PROJEKTA

- Perović Marija 10/22
- Đurišić Stefan 7/22
- Đuričić Đorđe 15/22
- Đerekarac Redžep 49/22

## G.1 IZVJEŠTAJ O PROBLEMU

Ovaj izvještaj opisuje proces razvoja softvera za igru Connect 4, koja omogućava igračima da se takmiče na računaru. Connect 4 je popularna igra strategije u kojoj dva igrača ili jedan igrač protiv računara pokušavaju postaviti četiri figure iste boje u vertikalnu, horizontalnu ili dijagonalnu liniju na 7x6 tabli.



***Slika:Primjer Connect 4***

Naša vizija uključuje implementaciju klasične verzije Connect 4 sa dodatnim funkcionalnostima radi poboljšanja korisničkog iskustva:

- **Singleplayer i multiplayer mod:** Omogućeno je takmičenje igrača protiv AI ili jedni protiv drugih, pružajući fleksibilnost i raznovrsnost u načinu igranja.
- **Grafički prikaz ploče:** Igračima se pruža vizuelni prikaz trenutnog stanja table tokom partije, što olakšava praćenje i planiranje poteza.
- **Pobjednički uslov:** Implementiran je mehanizam za provjeru i deklarisanje.

Za razvoj Connect 4 koristimo Python kao glavni programski jezik, uz dodatnu podršku Pygame biblioteke za implementaciju grafičkog korisničkog interfejsa. Pygame omogućava jednostavno upravljanje grafičkim elementima i interakciju sa korisnikom, što je ključno za realizaciju igre. Takođe koristimo tkinter za razvoj i implementaciju GUI-ja, numpy za pomoc pri backendu koda.

## G.1.2 RJEČNIK POJMOVA

1. **Tabla:** Igračka površina sa 7 kolona i 6 redova na koju se postavljaju figure.

2. **Figura:** Kružna figura koja predstavlja igrača na ploči.

3. **Igrač 1 / Igrač 2:** Dva igrača koji se takmiče u igri, označeni sa žutim i crvenim diskovima.

4. **AI (*Artificial Intelligence*):** Računarski protivnik koji simulira ponašanje igrača u singleplayer modu.

5. **Pobjednički uslov:** Stanje u kojem jedan igrač postavi četiri svoja diska u vertikalnu, horizontalnu ili dijagonalnu liniju.

6. **Remi:** Situacija u kojoj nema više dostupnih poteza za postizanje pobjedničkog uslova.

## G.2 INŽENJERING SISTEMSKIH ZAHTJEVA

Kroz ovaj proces, istražujemo i identifikujemo ključne probleme koje igra Connect 4 treba da riješi, kao što su efikasnost korisničkog interfejsa, upravljanje igračkim iskustvom, kao i implementacija naprednih funkcija poput AI za solo mod ili podrška za multiplayer.

## G.2.1 FUNKCIONALNI ZAHTJEVI I TRACEABILITY MATRICA

Pri izradi sistemskih zahtjeva za igru Connect 4, počinjemo sa definisanjem funkcionalnih zahtjeva na osnovu specifikacija korisnika i analize potreba sistema. Ovi zahtjevi se dalje razvijaju kroz iterativni proces kako bi se osigurala jasnoća, preciznost i ostvarivost resursa i vremenskog rasporeda projekta.

Nabrojani funkcionalni zahtjevi za igru Connect 4 su prikazani u sljedećoj traceability matrici

ID Zahtjeva	Naziv Zahtjeva	Opis Zahtjeva	Povezani Modul	Source Fajl	MAX PW	TOTAL PW
REQ-1	Prikaz glavnog menija	Program prikazuje glavni meni sa opcijama za pokretanje igre prikaz autora i izlaz.	root, start_game, show_authors, exit_program	start_game.py	5	3
REQ-2	Pokretanje igre	Korisnik može pokrenuti igru sa glavnog menija, birajući između singleplayer i multiplayer modova.	start_game, single_player, multiplayer	start_game.py	5	4
REQ-3	Prikaz autora	Program treba prikazati prozor sa imenima autora.	show_authors	start_game.py	2	2
REQ-4	Izlazak iz aplikacije	Korisnik može izaći iz programa klikom na dugme "Izlaz".	exit_program	start_game.py	2	1
REQ-5	Povratak na glavni meni	Nakon završetka igre, korisnik treba imati opciju da se vrati na glavni meni.	post_game_options, back_to_main	start_game.py	4	3
REQ-6	Ponovno pokretanje igre	Nakon završetka igre, korisnik treba imati opciju da pokrene novu partiju.	post_game_options, new_game, start_game	start_game.py	4	3
REQ-7	Prikaz igre sa jednim igračem	Program treba omogućiti igru protiv računara (AI).	single_player, connect4_with_ai.py	connect4_with_ai.py	5	4
REQ-8	Prikaz igre sa dva igrača	Program treba omogućiti igru za dva igrača.	multiplayer, connect4.py	connect4.py	5	4
REQ-9	Prikaz pobjednika	Program treba prikazati poruku kada jedan igrač pobjedi.	winning_move, label, screen.blit	connect4_with_ai.py, connect4.py	4	3
REQ-10	Prikaz nerjesenog rezultata	Program treba prikazati poruku kada se igra završi u remiju.	is_board_full, label, screen.blit	connect4_with_ai.py, connect4.py	4	3
REQ-11	Evaluacija poteza	Algoritam treba da evaluira najbolje poteze za računar (AI).	evaluate_window, score_position, minimax.get_valid_locations, pick_best_move	connect4_with_ai.py	5	4
REQ-12	Kreiranje table	Algoritam treba da kreira početnu praznu tablu.	create_board	connect4_with_ai.py, connect4.py	3	2
REQ-13	Crtanje table	Algoritam treba da ispiše trenutnu situaciju na tabli u konzoli.	print_board	connect4_with_ai.py, connect4.py	3	2
REQ-14	Provjera validnosti poteza	Algoritam treba da provjeri da li je potez validan tj. da li je kolona puna.	is_valid_location	connect4_with_ai.py, connect4.py	3	2
REQ-15	Pronalaženje sljedećeg slobodnog reda	Algoritam treba da pronade sljedeći slobodan red u koloni za ubacenu figuru.	get_next_open_row	connect4_with_ai.py, connect4.py	3	2
REQ-16	Ubacivanje figure	Algoritam treba da ubaci figuru u odgovarajuću kolonu i red na tabli.	drop_piece	connect4_with_ai.py, connect4.py	4	3
REQ-17	Prikaz novog stanja table	Program treba grafički da prikaže trenutnu situaciju na tabli.	draw_board, pygame.draw.rect, pygame.draw.circle	connect4_with_ai.py, connect4.py	4	3
REQ-18	Provjera pobjedničkog poteza	Algoritam treba da provjeri da li trenutni potez dovodi do pobjede igrača.	winning_move	connect4_with_ai.py, connect4.py	5	4
REQ-19	Provjera popunjenosti table	Algoritam treba da provjeri da li je tabla u potpunosti popunjena figurama.	is_board_full	connect4_with_ai.py, connect4.py	3	2

*Primjer Traceability Matrice*

Ovi funkcionalni zahtjevi su prioritetno rangirani prema njihovom uticaju na osnovne funkcije igre Connect 4 i krajnje korisničko iskustvo.

## **G.2.2 NABROJANI NEFUNKCIONALNI ZAHTJEVI ZA IGRU CONNECT 4**

U skladu sa funkcionalnim zahtjevima iz tabele G-1, nefunkcionalni zahtjevi za igru Connect 4 usredsređuju se na performanse, korisničko iskustvo i sigurnost sistema. Neki od ovih zahtjeva uključuju:

### **1. Performanse sistema:**

- Sistem treba da podrži glatko izvršavanje igre bez značajnog kašnjenja ili trzanja prilikom poteza igrača.
- Ovo je posebno važno za AI funkcionalnost kako bi se osiguralo brzo i efikasno donošenje odluka.

### **2. Korisničko iskustvo:**

- Grafički interfejs treba da bude intuitivan, jednostavan i privlačan, prilagođen korisnicima.
- Funkcija za ponovno igranje.
- Mogućnost biranja partije u singleplayer ili multiplayer modu.

### **3. Testiranje i verifikacija:**

- Implementirati detaljan plan testiranja kako bi se osiguralo da svi funkcionalni i nefunkcionalni zahtjevi zadovoljavaju očekivanja.

- Automatizacija testiranja kako bi se ubrzao proces verifikacije i smanjili potencijalni problemi tokom razvoja.

Nefunkcionalni zahtjevi igre Connect 4 pomažu u definisanju uslova pod kojima će sistem raditi i pružajući smjernice za optimizaciju performansi i korisničkog iskustva.

## G.2.3 TESTOVI PRIHVATANJA ZA IGRU CONNECT 4

**Testovi prihvatanja** su ključni za provjeru da li implementacija igre Connect 4 zadovoljava sve funkcionalne i nefunkcionalne zahtjeve. Izdovjeni su sljedeći zahtjevi, kao potencijalni:

### 1. Testiranje pokretanja same igre (*main menu sekcija*):

- **Scenario:** Korisnik pokreće video igru i otvara se prozor sa main menu-jem. On sadrži 3 opcije (*Zapocni igru, autori, izlaz*).
- **Očekivani rezultat:** Glavni meni se prikazuje sa zadatim opcijama i sve funkcionisu kako treba (tj. svako dugme je pointer odgovarajućem djelu koda)

### 2. Testiranje prikaza igre sa jednim igračem:

- **Scenario:** Korisnik započinje igru i bira **Singleplayer Mod.**
- **Očekivani rezultat:** Učitava se novi prozor sa praznom tablom. Korisnik (*Player 1*) je na potezu i program čeka njegov input.

### 3. Testiranje poteza igrača:

- **Scenario:** Korisnik odabira poziciju (*pomjeranjem miša po X osi*) u kojoj želi da ubaci figuru, i ubacuje (*pritiskom lijevog klika miša*) figuru.
- **Očekivani rezultat:** Figura se prikazuje na željenoj poziciji igrača, i program onda obavlja potez AI-ja, i onda se naizmjenično smjenjuju do kraja igre.

#### 4. Testiranje pobjedničkog poteza:

- **Scenario:** Player 1 postiže pobjednički potez i povezuje 4 figure u horizontalnu, vertikalnu ili dijagonalnu liniju.
- **Očekivani rezultat:** Igra završava sa porukom da je *Player 1* pobjednik, a nakon toga prikazuje se opcija za novu partiju ili povratak na glavni meni.

#### 5. Testiranje igre sa dva igrača:

- **Scenario:** Jedan od korisnika startuje program, započinje igru i bira **Multiplayer** mod.
- **Očekivani rezultat:** Igra se odvija naizmjenično između *Player-a 1* i *Player-a 2*. Svaki korisnik ima svoj potez, i tako se redom smjenjuju.

#### 6. Testiranje AI protivnika:

- **Scenario:** Korisnik odabire Singleplayer mod i igra protiv računara.
- **Očekivani rezultat:** AI protivnik donosi optimalne odluke prilikom izbora poteza, pružajući izazovanu igru korisniku.

#### 7. Testiranje poruke neriješenog rezultata:

- **Scenario:** Igra završava neriješenim rezultatom kada je tabla puna, a nema pobjednika.
- **Očekivani rezultat:** Prikazuje se poruka "DRAW!", sa opcijama za novu partiju ili povratak na glavni meni.

#### 8. Testiranje grafike i korisničkog interfejsa:

- **Scenario:** Korisnik interaguje sa grafikom table i izbornicima tokom igre.
- **Očekivani rezultat:** Svi elementi interfejsa (tabla, figure, dugmad) se pravilno prikazuju i reaguju na korisničke akcije bez problema.

Testovi prihvatanja igre Connect 4 su ključni za osiguravanje da igra zadovoljava sve definisane zahteve i pruža kvalitetno iskustvo korisnicima tokom svakog aspekta igre.

## G.3 SPECIFIKACIJA FUKCIONALNIH ZAHTJEVA

U ovom odjeljku, iako naslov sugerise specifikaciju zahtjeva, istražujemo i dalje sistemske funkcionalne zahteve za budući sistem, kao i preciziranje otkrivenih zahtjeva.

Počnemo odabirom arhitektonskog stila za naš sistem jer će korisničko iskustvo zavisiti od izbora arhitekture. Bez jasnih scenarija upotrebe, nemoguće je detaljno opisati arhitektonski stil koji će podržavati naše zahteve. U igri Connect 4, ključni sistemske zahtjevi obuhvataju:

1. **Arhitektonski stil:** Odabrali smo modularnu arhitekturu koja omogućava lakše dodavanje novih funkcionalnosti i održavanje sistema. Ovo uključuje module za korisnički interfejs, logiku igre i AI protivnika.
2. **Moduli sistema:**
  - **Korisnički interfejs:** Treba da podržava prikaz table, unos korisničkih poteza i prikaz rezultata igre.
  - **Logika igre:** Implementacija pravila Connect 4, validacija poteza, provjera pobjednika i neriješenog rezultata.



- **AI protivnik:** Modul koji omogućava računar da igra protiv korisnika, koristeći algoritam za donošenje optimalnih poteza.

### 3. Funkcionalni zahtjevi:

- **Pokretanje igre:** Sistem treba da omogući korisniku izbor između singleplayer i multiplayer moda, kao i pokretanje igre sa odgovarajućim interfejsom za odabir.
- **Igranje sa jednim igračem:** Implementacija AI protivnika koji pruža izazov igraču.
- **Igranje sa dva igrača:** Omogućavanje korisnicima da igraju jedan protiv drugog na istom računaru.
- **Prikaz pobjednika i neriješenog rezultata:** Sistem treba da jasno komunicira rezultate igre korisnicima nakon završetka partije.
- **Grafika i korisnički interfejs:** Prikaz table, figura, dugmadi i opcija interaktivnosti koje omogućavaju jednostavnu navigaciju i intuitivno korisničko iskustvo.

4. **Preciziranje zahtjeva:** Svaki zahtjev će biti detaljno analiziran i preciziran kako bi se osiguralo da ispunjava potrebe korisnika i da je izvodljiv u okvirima postavljenih projektnih resursa.

Specifikacije funkcionalnih zahtjeva za igru Connect 4 su ključne za dalji razvoj sistema, osiguravajući da svaki aspekt igre bude pažljivo razmatran i implementiran radi postizanja optimalnog korisničkog iskustva.

## **G.3.1 AKTERI**

U sistemu igre Connect 4, identifikujemo različite aktere koji imaju interes ili učestvuju u korišćenju ili razvoju sistema.

## **G.3.2 AKTERI I CILJEVI**

1. Player 1
2. Player 2
3. AI

## **G.3.3 USE CASES – POVREMENI OPISI**

### **Use Case: Pokretanje igre**

**Opis:** Ovaj Use Case opisuje proces pokretanja igre Connect 4.

**Glavni tok događaja:**

1. Igrač pokreće igru Connect 4.
2. Sistem prikazuje igračku tablu.

**Izvedeno iz zahtjeva:** REQ1, REQ2

### **Use Case: Ubacivanje figure u odabranu kolonu**

**Opis:** Ovaj Use Case opisuje proces ubacivanja figura u odabranu kolonu igračke table.

**Glavni tok događaja:**

1. Igrač odabire kolonu za ubacivanje figure.
2. Sistem provjerava validnost odabrane kolone.
3. Sistem ubacuje figuru u odgovarajući red kolone.

**Izvedeno iz zahtjeva:** REQ16, REQ17

### **Use Case: Provjera pobjedničkog poteza**

**Opis:** Ovaj Use Case opisuje proces provjere da li je igrač napravio potez koji ga dovodi do pobjede.

**Glavni tok događaja:**

1. Nakon svakog poteza igrača, sistem provjerava da li je novi potez pobjednički.
2. Ako je potez pobjednički, sistem prikazuje poruku o pobjedniku i završava igru.

**Izvedeno iz zahtjeva:** REQ18, REQ19

### **Use Case: Završetak igre**

**Opis:** Ovaj Use Case opisuje proces završetka igre i prikazivanja konačnog rezultata.

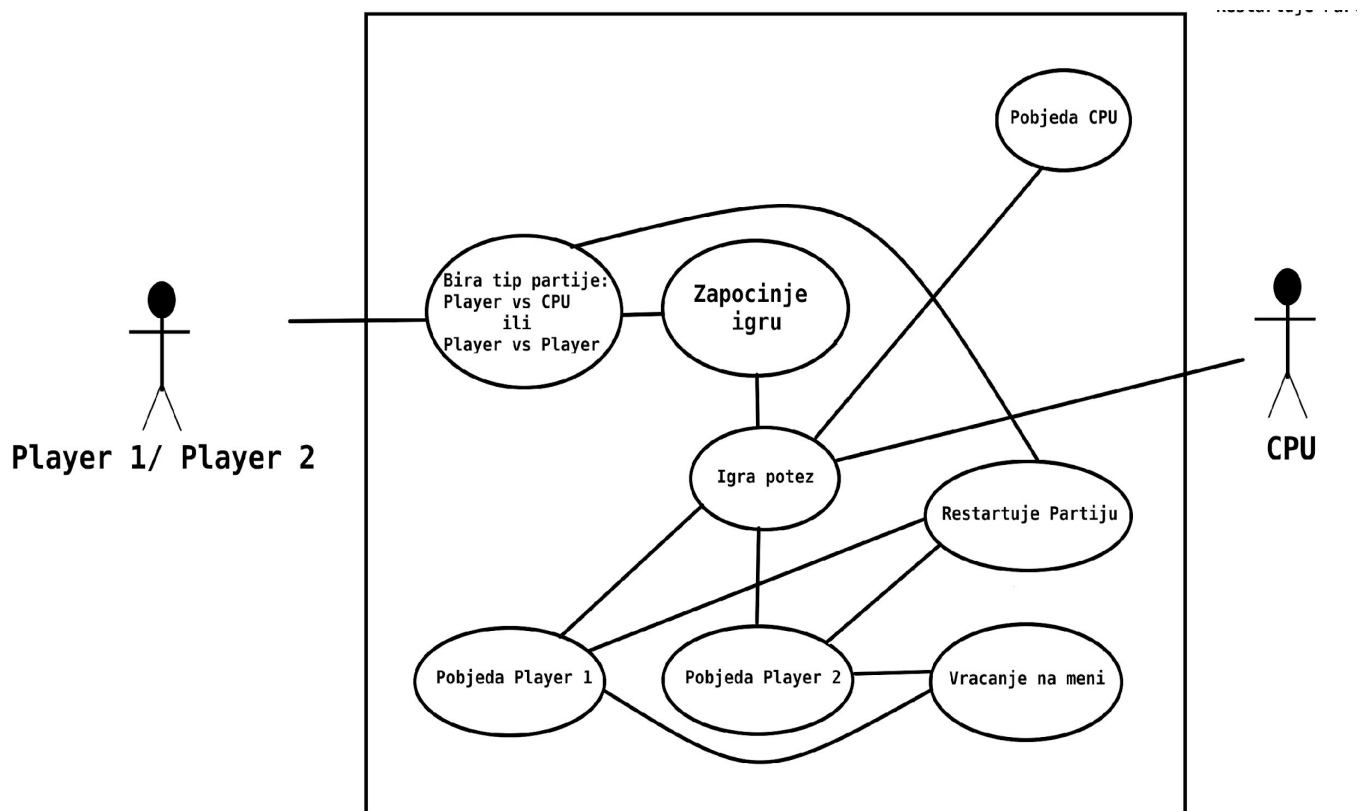
**Glavni tok događaja:**

1. Igrač uzima pobjednički potez ili sva polja table bivaju popunjena.
2. Sistem završava igru i prikazuje konačni rezultat na ekranu.
3. Prozor igrice se zatvara, i igrice otvara novi prozor sa opcijama "Igraj još jednu partiju" i "Povratak na glavni meni".

## Izvedeno iz zahtjeva: REQ7

Ovi Use Case-ovi opisuju osnovne interakcije koje se očekuju u igri Connect 4, pokrivajući procese kao što su pokretanje igre, odabir poteza, ubacivanje figura, provjera pobjedničkog poteza i završetak igre. Svaki od njih je direktno povezan sa specifičnim zahtjevima (REQ) koji su identifikovani u REQ tabeli.

### USE CASE DIAGRAM



Use Case ID	Naziv Use Case-a	Opis	Vezani zahtjeviP	Prioriteti
UC-1	Pokretanje igre Connect 4	Pokretanje aplikacije za igru Connect 4, inicijalizacija tabele i prikaz glavnog menija	REQ-1 REQ-8	Visok
UC-2	Potez	Proces odabira kolone za ubacivanje diska od strane igrača.	REQ-2 REQ-3 REQ-4 REQ-5	Visok
UC-3	Prikazi mogućih poteza	Prikazivanje mogućih kolona u koje se može ubaciti disk nakon odabira igrača.	REQ-6 REQ-7	Srednji
UC-4	Izvršavanje poteza	<div> <b>VALIDACIJA, UBACIVANJE DISKA U ODABRANU KOLONU I AŽURIRANJE STANJA TABLE.</b> </div>	REQ-7	Visok

UC-5	Zaustavljanje pobjede	<b>PROCES  OGRANIČAV  ANJA  IGRAČA U  NJIHOVIM  POTEZIMA  NAKON ŠTO  JEDAN  IGRAČ  POSTIGNE  POBJEDNIČK  I USLOV.</b>	REQ-6	Srednji
UC-6	Ponovno pokretanje igre	<b>OMOGUĆAV  ANJE  KORISNIKU  DA PONOVO  POKRENE  IGRU NAKON  ZAVRŠETKA  PARTIJE.</b>	REQ-8	Srednji
UC-7	Završetak igre	<b>ZAVRŠETAK  IGRE NAKON  ŠTO JE  JEDAN  IGRAČ  POSTIGAO  POBJEDU ILI  KADA JE  TABLICA  POTPUNO  POPUNJENA  NERIJEŠENO  .</b>	/	Visok

## **G.3.5 TESTOVI PRIHVATLJIVOSTI ZA USE CASES**

### **Test prihvatljivosti za UC-1: Pokretanje nove igre Connect 4**

**Testiranje slučaja upotrebe:** Pokretanje nove igre Connect 4

**Kriterijumi za prolaz/neuspjeh:** Igra se uspješno pokreće i korisnik može odmah započeti igru.

**Unos podataka:** Nije potreban unos podataka.

#### **Procedura testiranja:**

1. Pokrenuti aplikaciju za Connect 4.
2. Proveriti da se prikazuje igračka tabla dimenzija 7x6 sa 42 polja.
3. Proveriti da su polja prazna (bez igračkih figura).
4. Proveriti da su igrači pravilno identifikovani (npr. igrač 1 sa crvenim figurama, igrač 2 sa žutim figurama).
5. Proveriti da je igra spremna za početak (korisnik može odabrati kolonu za igranje).

#### **Očekivani rezultat:**

- Prikazuje se igračka tabla dimenzija 7x6 sa 42 polja.
- Polja su prazna i spremna za postavljanje figura.
- Igrači su identifikovani sa odgovarajućim bojama figura.
- Korisnik može odabrati kolonu za igranje kako bi započeo igru.

**Testiranje slučaja upotrebe:** Postavljanje figura u Connect 4 igri

**Kriterijumi za prolaz/neuspeh:** Žetoni se uspešno postavljaju u odabranu kolonu i ažurira se stanje igre.

**Unos podataka:** Nije potreban unos podataka.

**Procedura testiranja:**

1. Pokrenuti aplikaciju za Connect 4.
2. Odabrati kolonu u koji želite postaviti svoju figuru (npr. stub 3).
3. Proveriti da se žeton pravilno postavlja u odabrani stub.
4. Proveriti da se igrački red ažurira i prelazi na sledećeg igrača nakon postavljanja žetona.
5. Ponoviti korake 2-4 sa različitim stubovima za postavljanje žetona.

**Očekivani rezultat:**

- Figura se pravilno postavlja u odabrani stub.
- Stanje igre se ažurira nakon postavljanja figura (red igrača menja).
- Igra omogućava nastavak sa sledećim potezom nakon postavljanja figura.

**Identifikator probnog slučaja:** TC003

**Testiranje slučaja upotrebe:** Provera pobednika u Connect 4 igri

**Kriterijumi za prolaz/neuspeh:** Igra uspešno identifikuje pobednika ili remi situaciju.

**Unos podataka:** Nije potreban unos podataka.

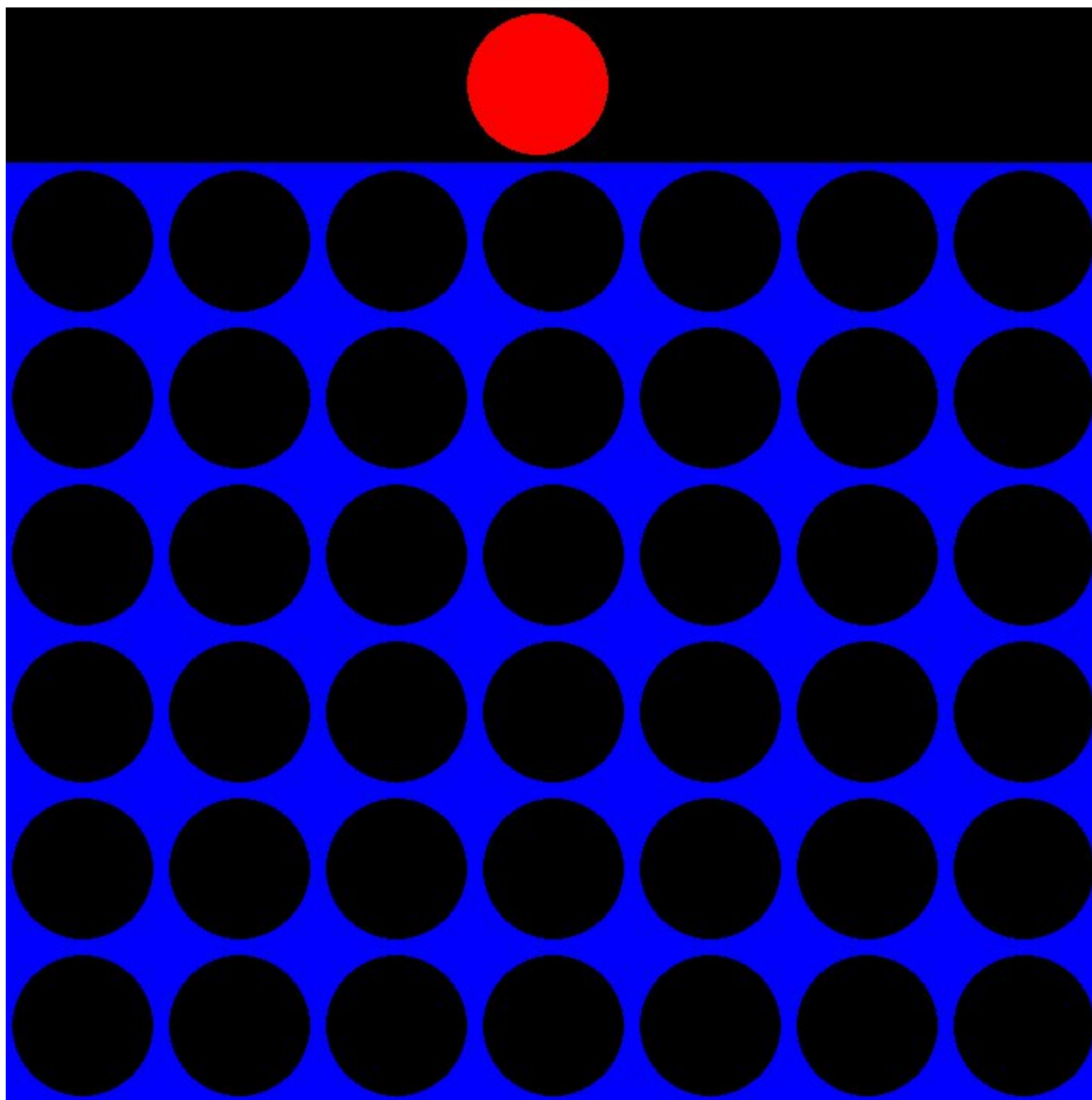


## **Procedura testiranja:**

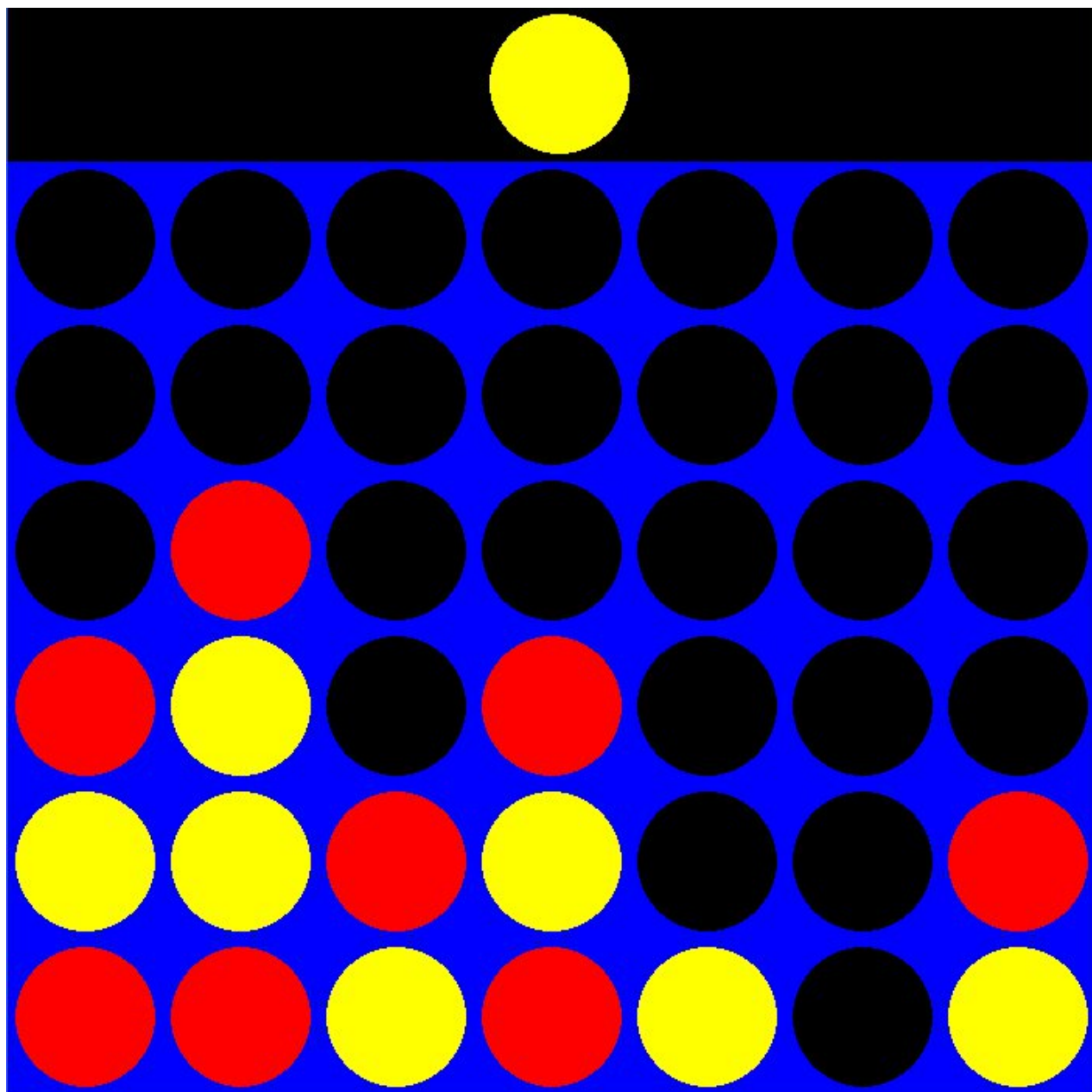
1. Pokrenuti aplikaciju za Connect 4.
2. Igrati partiju gde se postavljaju figure kako bi se formirala linija od 4 istih boja (horizontalno, vertikalno ili dijagonalno).
3. Proveriti da se igraču koji prvi formira liniju od 4 figure iste boje priznaje pobeda.
4. Proveriti da se igra završava remijem ukoliko nijedan igrač ne formira liniju od 4 figure iste boje.
5. Ponoviti korake 2-4 sa različitim situacijama za formiranje linije.

## **Očekivani rezultat:**

- Igra uspješno identifikuje pobjednika ako se formira linija od 4 figure iste boje.
- Igra završava remijem ako nijedan igrač ne formira liniju od 4 figure iste boje.

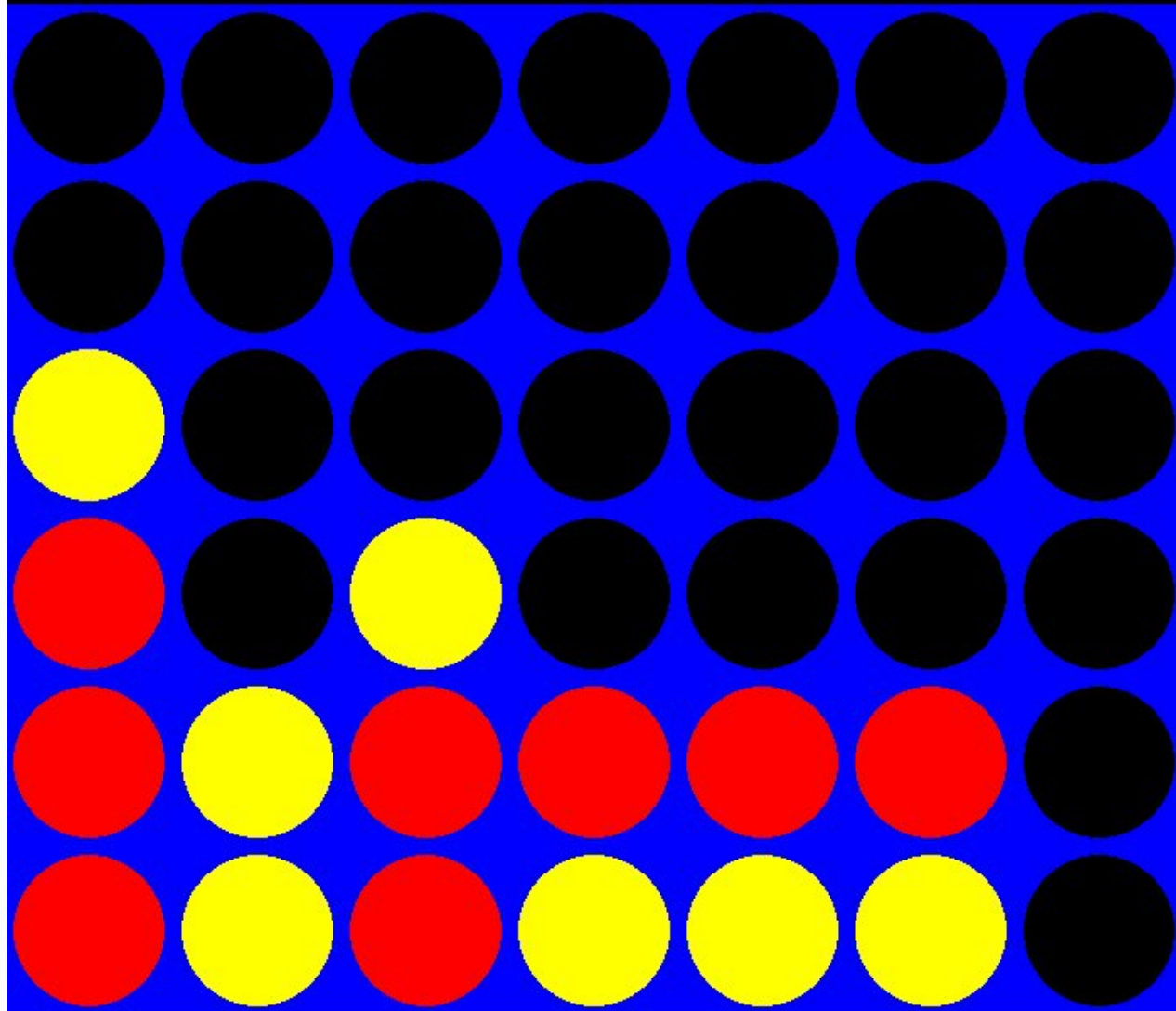


*Primjer-početna faza igre(prazna tabla, čeka Player 1 move)*



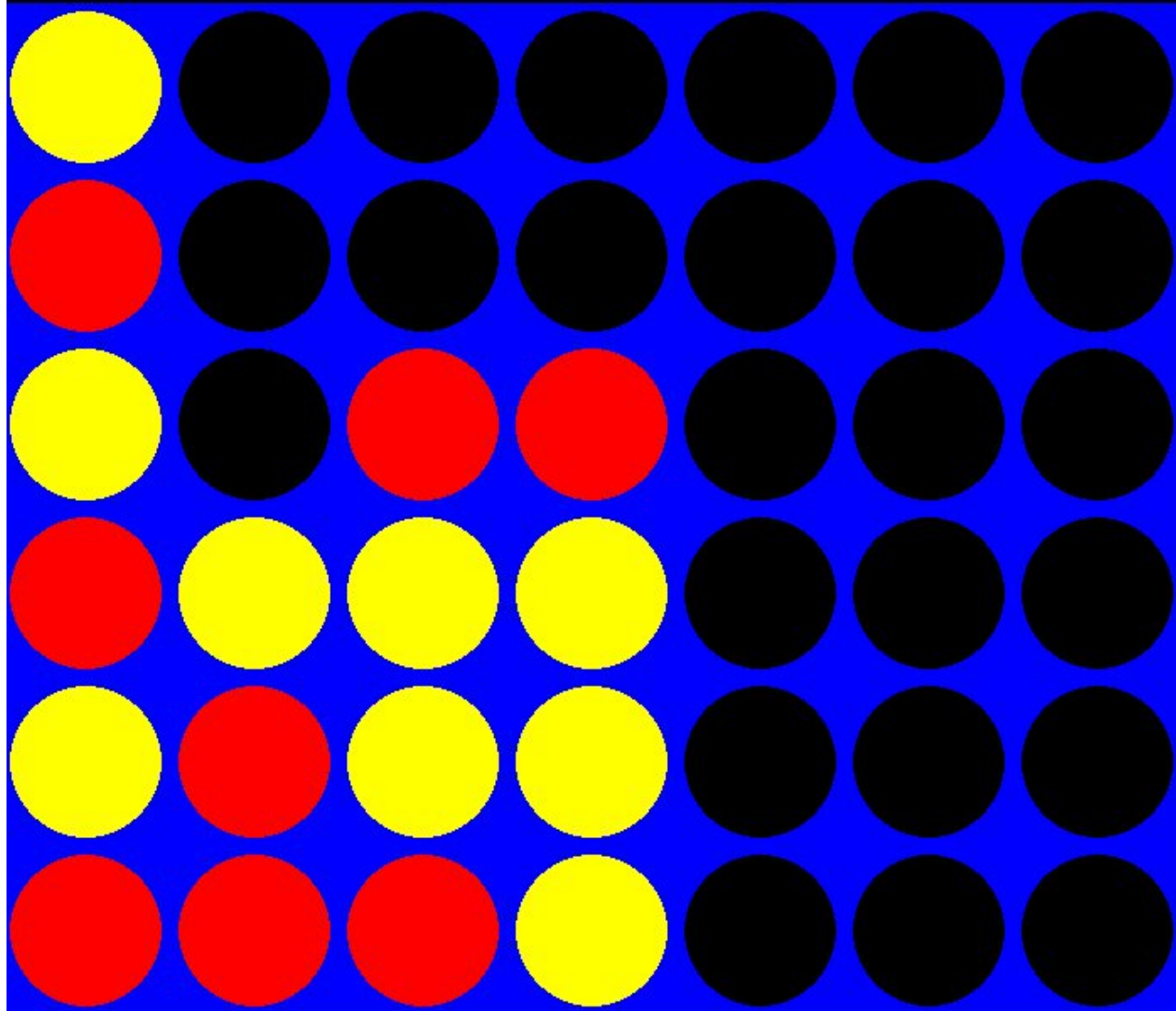
*Primjer-faze tokom igranja igre (midplay, čeka Player 2 move)*

PLAYER 1 WINS!



*Primjer-faza završetka igre (Player 1 Win)*

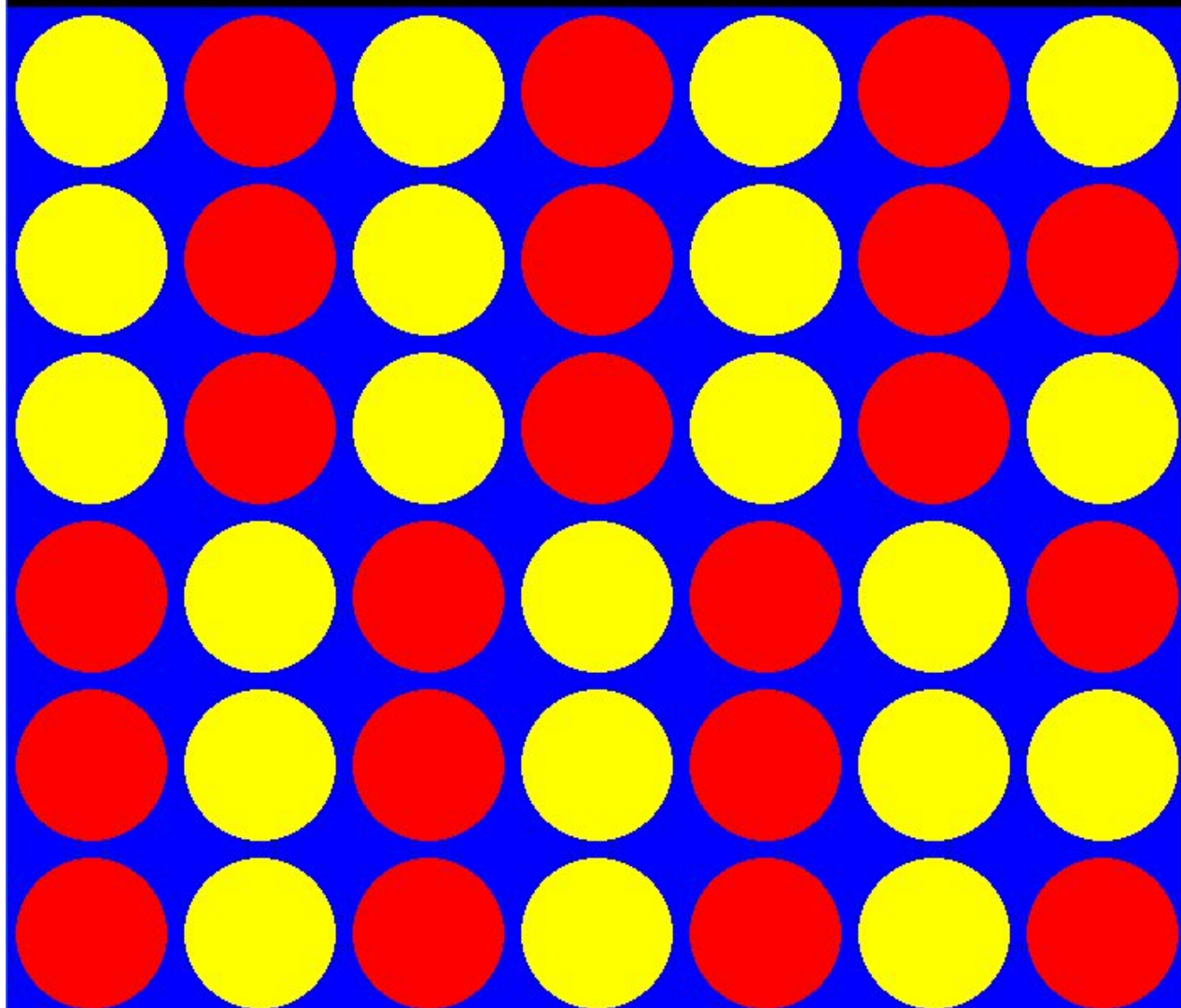
PLAYER 2 WINS!



*Primjer-faza završetka igre (Player 2 Win)*

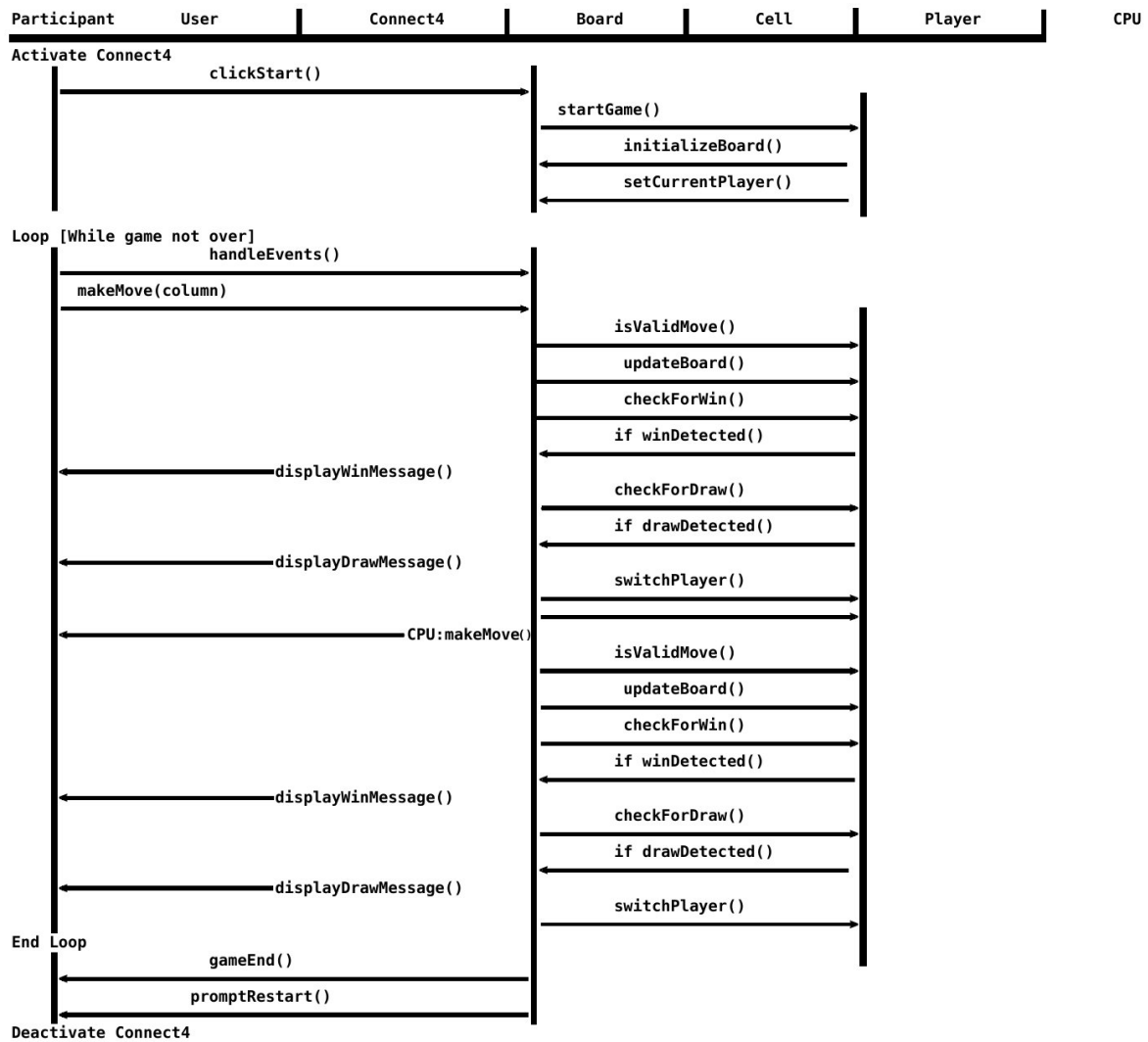


DRAW !



*Primjer-faza završetka igre (Remi/Draw)*

## G.3.6 SISTEMSKI DIJAGRAM SEKVENCI



## G.4 SPECIFIKACIJE KORISNIČKOG INTERFEJSA

Ovaj odjeljak prikazuje našu skicu, koja pokazuje kako smo predvidjeli pojavljivanje korisničkog interfejsa na ekranu.

### G.4.1 Preliminarni UI dizajn

Za dati slučaj upotrebe, pokažimo korak po korak kako korisnik pravi poteze i kako potezi se pojavljuju na ekranu u igri Connect 4.

#### 9. Pokretanje igre

- o Korisnik pokreće aplikaciju za Connect 4.
- o Prikazuje se početni ekran sa opcijama: "Singleplayer" i "Multiplayer"

#### 10. Izbor moda igre

- o Korisnik bira mod igre (npr. "Singleplayer").

#### 11. Prikaz ploče za igru

- o Prikazuje se tabla sa 7 kolona i 6 redova.
- o Svaki igrač ima svoje figure različitih boja (npr. crveni i žuti).

#### 12. Odabir kolone za potez

- o Korisnik klikne na željenu kolonu gdje želi da postavi figuru.
- o Figura pada na najniže dostupno mjesto u izabranoj koloni.

#### 13. Validacija i animacija poteza

- o Sistem provjerava validnost poteza.
- o Ako je potez validan, figura ispunjava najniže dostupno mjesto u izabranoj koloni.
- o Ažurira se prikaz ploče sa novim stanjem.

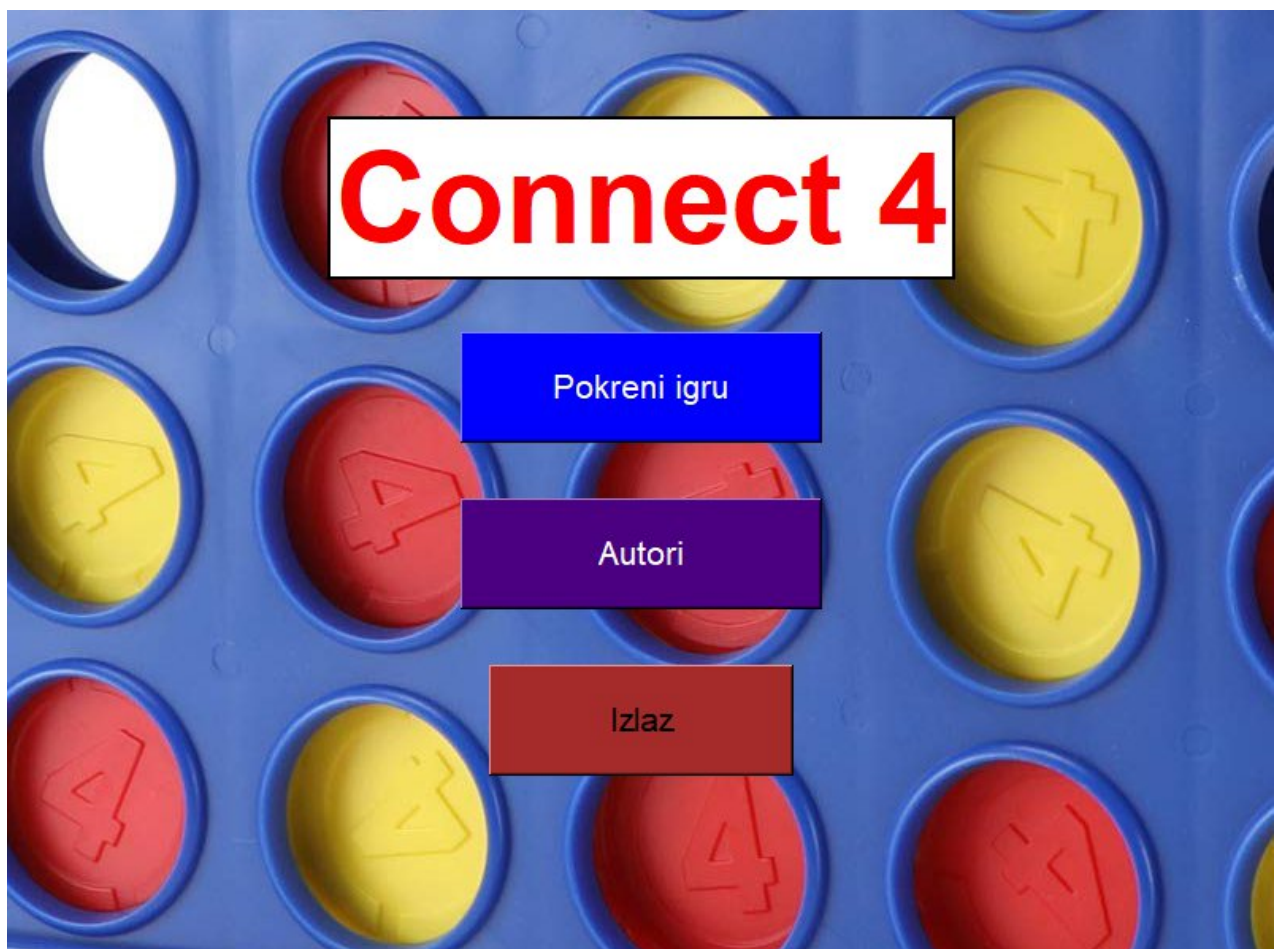
#### 14. Provjera pobjedničkog uslova

- o Nakon svakog poteza, sistem provjerava da li ima četiri figure iste boje u vertikalnoj, horizontalnoj ili dijagonalnoj liniji.
- o Ako postoji pobjednički uslov, prikazuje se poruka sa informacijom o pobjedniku.



- o Ako nema pobjedničkog uslova, igra se nastavlja.
- 15. Naredni potez**
  - o Ako igrača igra protiv AI, nakon korisnikovog poteza, AI pravi svoj potez.
  - o Ako je "Multiplayer" mod, drugi igrač bira kolonu za svoj potez.
- 16. Kraj igre**
  - o Kada je ispunjen pobjednički uslov ili je ploča puna, prikazuje se završni ekran sa opcijama: "Igraj još jednu partiju" i "Povratak na glavni meni".

## Prikaz preliminaranog dizajna korisničkog interfejsa



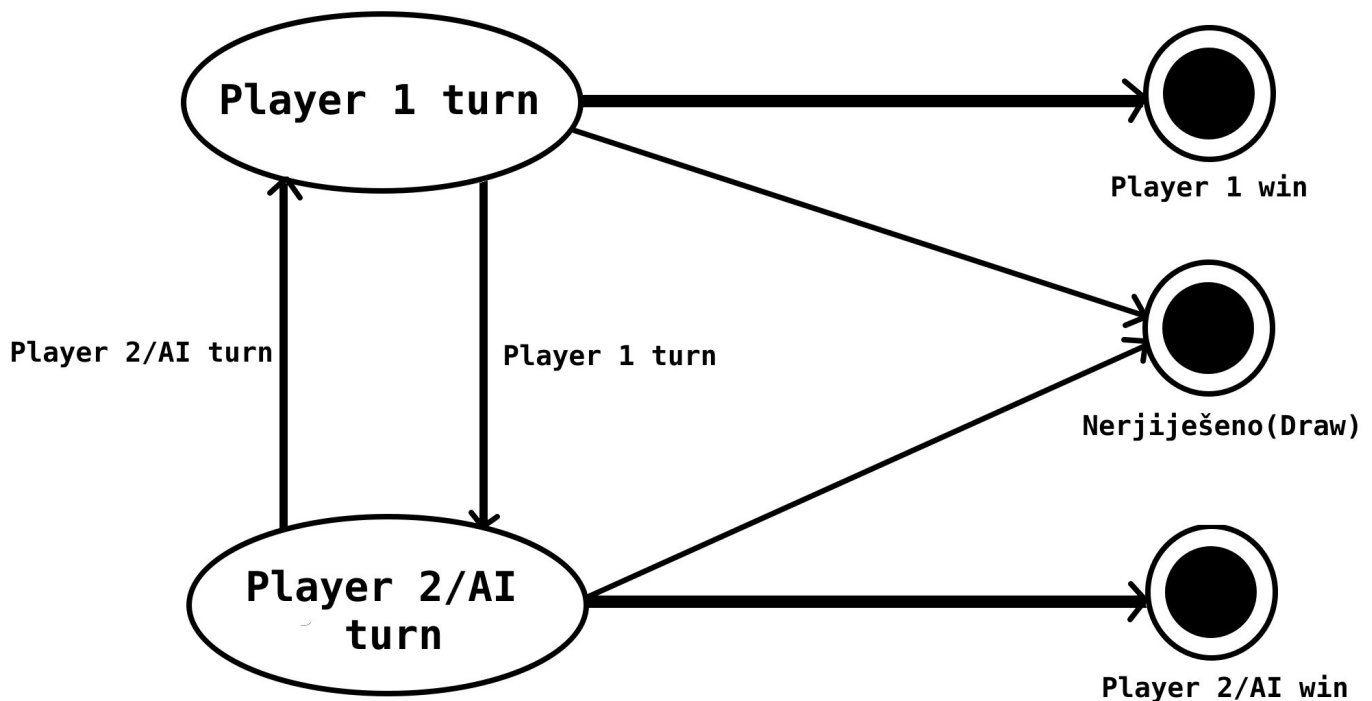
*Slika-Preview Glavnog menija*

- **Početni ekran**
  - o Opcije za izbor moda igre.
  - o Opcija za pregled autora
  - o Opcija za izlaz
- **Ekran igre**
  - o Tabela sa 7x6 polja.
  - o Prikaz figura i boja igrača.
  - o Informacije o trenutnom potezu (ko je na potezu).
- **Ekran za pobjedu/poraz**
  - o Prikazuje se poruka o pobjedniku.
  - o Opcije "Igraj još jednu partiju" i "Povratak na glavni meni".



***Primjer-Selekcija gamemod-a***

## G.5. STATE DIJAGRAM



### Početno stanje

-Početno stanje igre predstavlja praznu tabelu i čeka se potez Player-a 1

### Player 1 na potezu

-Stanje predstavlja situaciju kad je Player 1 na potezu. Player 1 bira poziciju na kojoj želi ubaciti figuru. U slučaju ako je neki red/kolona popunjen/a (tj, nema slobodnog polja za ubacivanje figure), igrač neće moći da ubaci figuru na toj poziciji, već će morati da obavi svoj potez na nekoj drugoj slobodnoj poziciji.

## Player 2/AI

-Što se tiče Player-a, stanje je slično kao i kod Player-a 1. Ista pravila važe i za AI, samo što za razliku od Player-a 2 koji ručno bira poziciju, AI ima svoj algoritam po kome postavlja figuru u tabeli. Bilo da je to ređanje figura da bi se osigurala pobjeda, ili blokiranje stack-a figura Player-a 1 da uradi slično (doda još jednu figuru kako bi došao do pobjede).

## Player 1 Win

-Predstavlja završetak igre pobjedom Player-a 1. Do ovoga dolazi kada Player 1 uspije da spoji 4 figure u bilo kome pravcu.

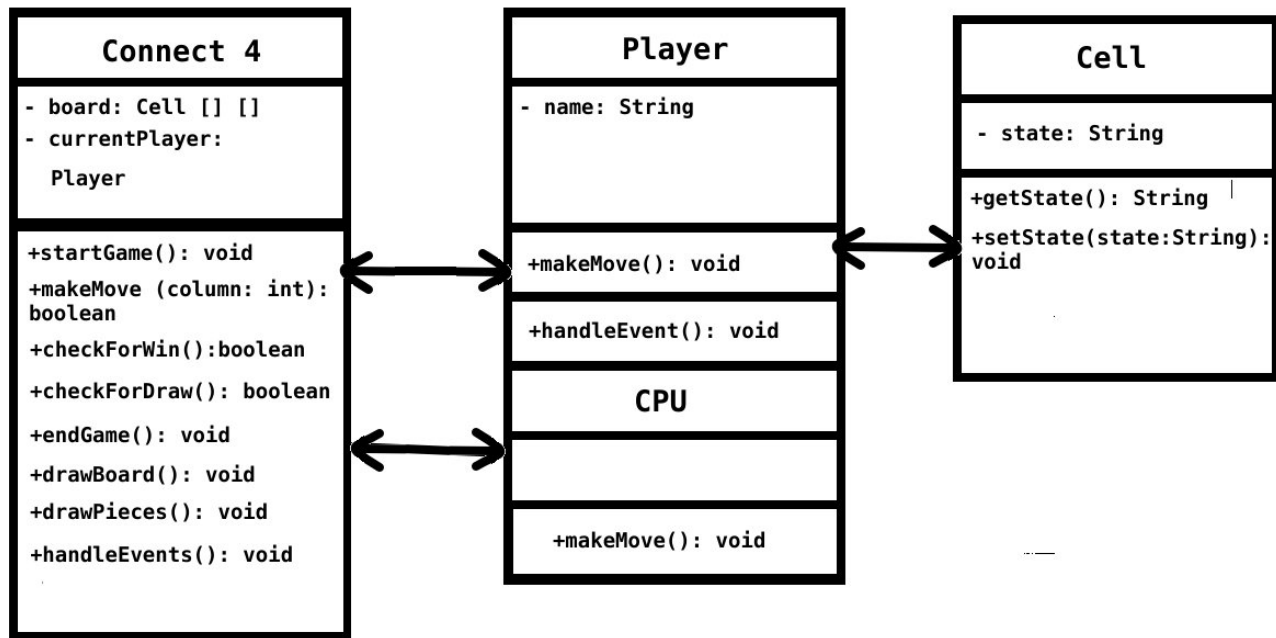
## Player 2/AI Win

-Slično stanje kao kod Player 1 wina samo što je ovdje pobjednik Player 2, odnosno AI (ako se igra singleplayer mod).

## Neriješeno (Draw)

-Do ovoga stanja dolazi, pri popunjavanju svih slobodnih polja tabele, pri čemu nijedan igrač (Player 1, Player 2 ili AI) nije bio u mogućnosti da složi 4 figure uzastopno.

## G.6. DIJAGRAM KLASA



## Uloge:

- 1.Redžep Đerekarac-dizajniranje, uređivanje i kreiranje dijagrama, saradnja pri uređivanju i pisanju dokumentacije, backend kod, doprinos AI developmentu, QA i game testing (platforma Linux), Linux build
2. Đorđe Đuričić-saradnja pri uređivanju i pisanju dokumentacije, backend kod, doprinos AI developmentu, QA i game testing (platforma Windows)
- 3.Stefan Đuričić-izrada i dizajniranje GUI-a, frontend kod, graficki dizajn, testiranje i uređivanje GUI komponenti
- 4.Marija Perović-saradnja pri uređivanju i pisanju dokumentacije, testiranje i uređivanje GUI komponenti koda