

# Softversko inženjerstvo

-dokumentacija mini-projekta-

## Tema: Memory Game

Projekat radili:

1. Tijana Bošković 59/23
2. Nikolina Fatić 63/23
3. Andrej Kastratović 83/23
4. Luka Tomanović 39/23

Podgorica, 10.06.2025. godine

## Kratak opis projekta:

Cilj ovog projektnog zadatka bio je da se implementira edukativna i zabavna igra memorije koristeći programski jezik **Python** i GUI biblioteku **Tkinter**. Igra omogućava korisniku da razvija pažnju i pamćenje kroz pronalaženje parova skrivenih kartica.

Korisniku su ponuđeni različiti nivoi, od lakog (2x2) do teškog (6x6), uključujući i specijalne nivoe kao što su:

- Hard Mode sa vremenskim ograničenjem,
- Blind Mode, gdje su kartice prikazane na kratko, nakon čega igrač mora igrati "naslijepo".

Dodana su funkcionalna poboljšanja kao što su:

- brojanje pokušaja,
- vizuelno otkrivanje i skrivanje kartica,
- automatski povratak na početni ekran nakon pobjede ili isteka vremena.

## Opis zadataka i doprinosa članova projektnog tima

### Nikolina Fatic:

Kreiranje klase, prikaz pocetnog ekrana kao i funkcionisanje kartica. Objašnjenje koda.

### Tijana Boskovic:

Kreiranje klase, kao i kreiranje modova(hard mod i blind mod). Objašnjenje koda.

### Andrej Kastratovic:

Izrada dokumentacije. Izrada funkcionalnih zahtjeva, testova prihvatanja, use case-ova.

### Luka Tomanovic:

Izrada use case dijagrama, kao i dijagrama sekvence i aktivnosti.

## Opis opštih funkcionalnosti softvera

Igra Memory Game je klasična logička igra u kojoj je cilj pronaći sve parove istih brojeva skrivenih iza kartica.

Osnovna pravila:

1. Igra se odvija na mreži dimenzija 2x2, 4x4 ili 6x6, u zavisnosti od izabranog nivoa.
2. Svaka kartica krije broj. Igrač klikom otkriva dvije kartice.
3. Ako su brojevi isti, par ostaje otkriven.
4. Ako nisu isti, kartice se zatvaraju nakon kratkog vremena.
5. Igra se nastavlja dok se svi parovi ne pronađu.

Specijalni režimi:

- **Hard Mode:**  
Igra traje 60 sekundi. Ako ne pronađeš sve parove na vrijeme – gubiš.
- **Blind Mode:**  
Kartice su prikazane na 3 sekunde na početku, nakon čega se igra nastavlja bez vizuelnog otkrivanja vrijednosti.

Cilj igre:

Pronaći sve parove sa što manje pokušaja, a u nekim režimima i u što kraćem vremenskom roku.

## Funkcionalni zahtjevi

Identifikator	PW	Zahtjev
Req1	5	Sistem prikazuje početni ekran sa naslovom i dugmetom za početak igre.
Req2	5	Sistem omogućava izbor između više nivoa težine (2x2, 4x4, 6x6, Hard Mode, Blind Mode).
Req3	5	Sistem nasumično raspoređuje parove brojeva na polja i omogućava okretanje kartica.
Req4	2	Sistem pamti broj pokušaja i prikazuje ga tokom igre.
Req5	3	U režimu "Hard Mode", uključuje se tajmer koji odbrojava vrijeme.
Req6	3	U režimu "Blind Mode", kartice se kratko otkriju na početku i zatim igra u potpunosti bez vidljivih vrednosti.
Req7	5	Kada se svi parovi pronađu, prikazuje se poruka o pobjedi i korisnik se vraća na izbor nivoa.
Req8	4	Dugme za povratak na meni omogućava restart igre bez gašenja aplikacije.

## Slučajevi testa prihvatanja

- REQ1 – Početni ekran sa naslovom i dugmetom

### Koraci:

1. Pokreni igru.
2. Provjeri da li se prikazuje naslov "Memory Game".
3. Provjeri da li postoji dugme "Počni igru".

- REQ2 – Odabir nivoa težine

### Koraci:

1. Klikni na "Počni igru".
2. Provjeri da li se prikazuju svi ponuđeni nivoi:
  - Lako (2x2)
  - Srednje (4x4)
  - Teško (6x6)
  - Hard Mode
  - Blind Mode
3. Klikom na bilo koji nivo, provjeri da li igra kreće sa odgovarajućom veličinom table.

- REQ3 – Kartice se miješaju i mogu se okretati

**Koraci:**

1. Odaberi bilo koji nivo (npr. 4x4).
  2. Klikni na bilo koje dvije kartice.
  3. Proveri da li se vrijednosti otkrivaju.
  4. Provjeri da li, u slučaju para, kartice ostaju otkrivene; ako nije par – ponovo se zatvaraju.
- REQ4 – Prikaz broja pokušaja

**Koraci:**

1. Odaberi nivo i započni igru.
  2. Okreni dvije kartice.
  3. Provjeri da li se broj pokušaja povećao za 1 i prikazuje pored teksta "Pokušaji".
- REQ5 – Tajmer u Hard Mode režimu

**Koraci:**

1. Izaberi nivo "Hard Mode".
  2. Provjeri da li se prikazuje tajmer sa 60s.
  3. Sačekaj 60 sekundi bez pronalaženja svih parova.
  4. Provjeri da li se pojavljuje poruka "Vrijeme isteklo".
- REQ6 – Blind Mode

**Koraci:**

1. Izaberi "Blind Mode".
  2. Provjeri da li su sve kartice kratko prikazane (oko 3s).
  3. Nakon toga, provjeri da li se karte zatvaraju i više se ne otvaraju klikom (osim prilikom pokušaja para).
  4. Kada klikneš dva puta, proveri da li sistem upoređuje interne vrijednosti i samo potvrđuje parove.
- REQ7 – Završetak igre i povratak na meni

**Koraci:**

1. Igraj dok ne pronađeš sve parove.
2. Provjeri da li se pojavljuje poruka o pobjedi (sa brojem pokušaja).
3. Provjeri da li te sistem vraća na meni za izbor nivoa.

- REQ8 – Dugme za povratak

**Koraci:**

1. U bilo kom trenutku igre, klikni na dugme "Nazad na izbor nivoa".
2. Provjeri da li se vratiš na ekran sa izborom nivoa.

## Use Case

Use Case: Pokretanje igre memorije

**Opis:**

Ovaj Use Case opisuje proces pokretanja igre, uključujući prikaz početnog ekrana, izbor nivoa težine i pripremu kartica.

**Glavni tok događaja:**

- Korisnik pokreće igru.
- Sistem prikazuje početni ekran sa dugmetom za početak.
- Nakon pritiska, korisniku se nudi izbor nivoa težine.
- Po izboru nivoa, sistem prikazuje mrežu kartica i pokreće igru.

**Izvedeno iz zahtjeva:** REQ1

Use Case: Izbor kartice i pokušaj pronalaska para

**Opis:**

Ovaj Use Case opisuje korisnikov pokušaj da pronađe dva ista broja klikom na kartice.

**Glavni tok događaja:**

- Korisnik klikne na jednu karticu – kartica se otkriva.
- Korisnik zatim klikne na drugu karticu.
- Ako su brojevi isti, kartice ostaju otkrivene.
- Ako nisu, kartice se vraćaju u početno stanje nakon kratkog vremena.

**Izvedeno iz zahtjeva:** REQ2

Use Case: Brojanje pokušaja i praćenje igre

**Opis:**

Ovaj Use Case opisuje sistemsko brojanje pokušaja korisnika i označavanje toka igre.

**Glavni tok događaja:**

- Sistem pamti svaki par klikova (pokušaj).

- Broj pokušaja se prikazuje u realnom vremenu na ekranu.

**Izvedeno iz zahtjeva:** REQ3

Use Case: Prikaz svih kartica na početku (Blind/Hard Mode)

**Opis:**

Ovaj Use Case opisuje kako se u specijalnim režimima igre sve kartice kratko otkriju na početku.

**Glavni tok događaja:**

- Po pokretanju igre u "Blind" ili "Hard" režimu, sistem prikazuje sve kartice na 3 sekunde.
- Zatim ih automatski zatvara i igra može da počne.

**Izvedeno iz zahtjeva:** REQ4

Use Case: Tajmer i ograničeno vreme (Hard Mode)

**Opis:**

Ovaj Use Case opisuje proces odbrojavanja vremena kada je aktiviran režim sa tajmerom.

**Glavni tok događaja:**

- Po pokretanju igre u "Hard Mode", sistem uključuje tajmer (npr. 60 sekundi).
- Tajmer se prikazuje u gornjem dijelu ekrana.
- Ako vrijeme istekne pre nego što su svi parovi pronađeni, igra se automatski završava porukom.

**Izvedeno iz zahtjeva:** REQ5

Use Case: Pobjeda u igri

**Opis:**

Ovaj Use Case opisuje završetak igre kada korisnik pronađe sve parove.

**Glavni tok događaja:**

- Korisnik pronađe sve parove kartica.
- Sistem prikazuje poruku o pobjedi, zajedno sa brojem pokušaja.
- Igra se resetuje ili se korisnik vraća na izbor nivoa.

**Izvedeno iz zahtjeva:** REQ6, REQ7

Use Case: Ponovno pokretanje igre

**Opis:**

Ovaj Use Case opisuje ponovni početak igre putem dugmeta.

**Glavni tok događaja:**

- Korisnik klikne na dugme "← Nazad na izbor nivoa".



- Sistem briše trenutno stanje i prikazuje izbor nivoaa.  
**Izvedeno iz zahtjeva:** REQ8

## Use Case dijagram

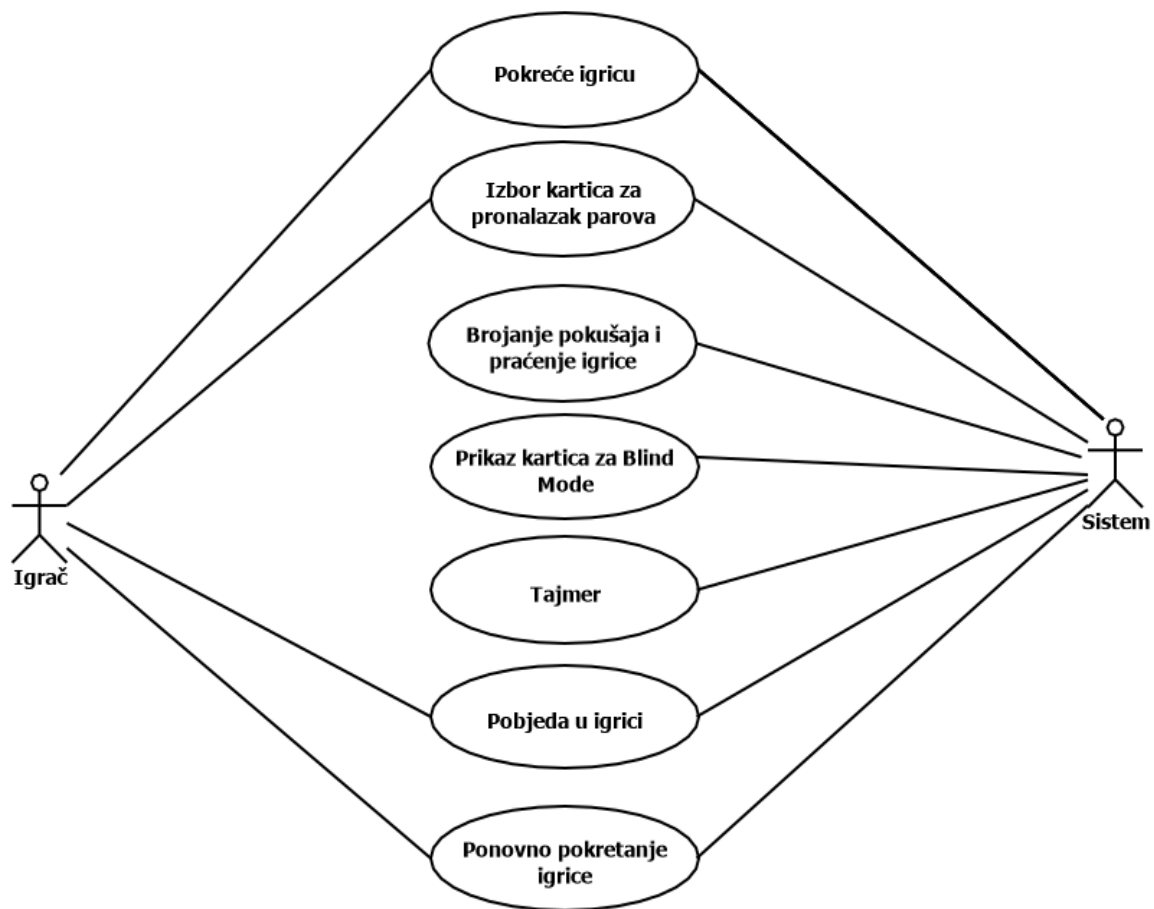
Use Case dijagram prikazuje osnovne funkcionalnosti aplikacije, kroz perspektivu korisnika sistema. Dijagram služi za vizuelno predstavljanje odnosa između korisnika i glavnih slučajeva upotrebe (use case-ova), koji definišu ponašanje sistema na najvišem nivou apstrakcije.

Cilj dijagrama je da jasno ilustruje sve dostupne interakcije koje korisnik može ostvariti sa aplikacijom, uključujući izbor nivoaa, igru, praćenje vremena, ponovno pokretanje igre i pregled rezultata.

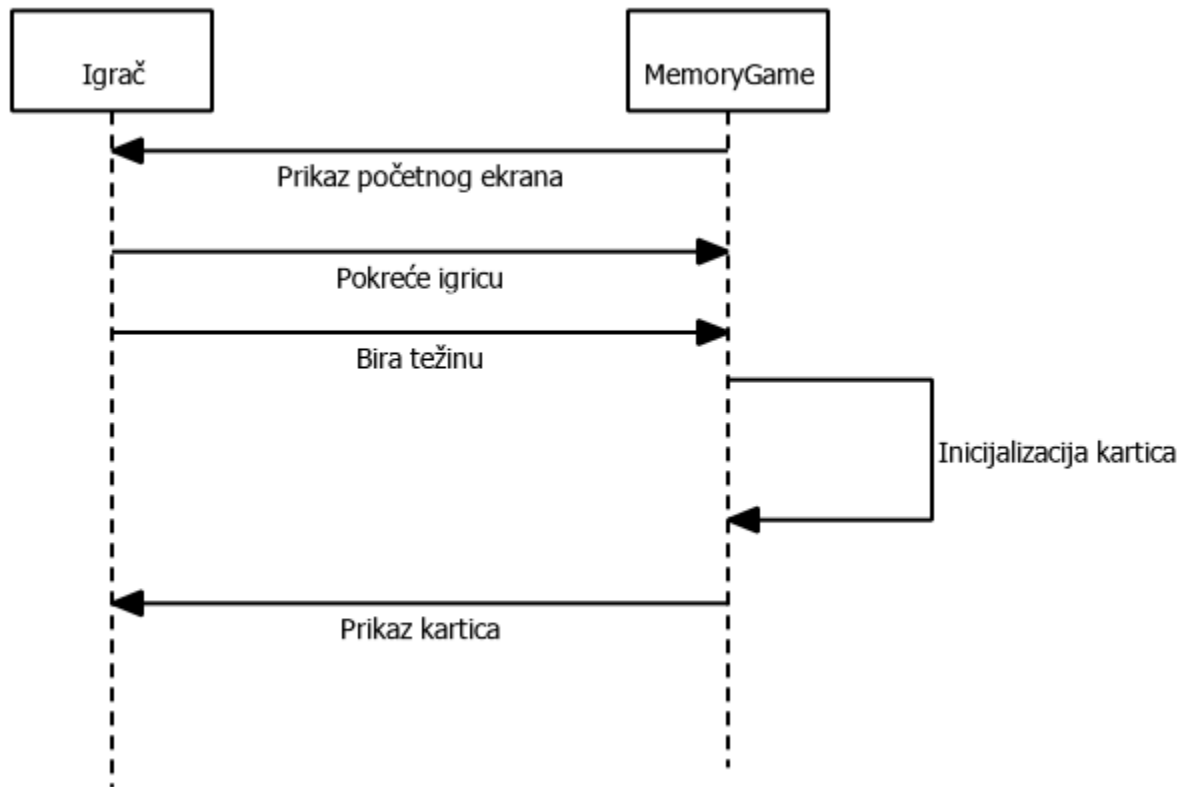
Dijagram uključuje sledeće funkcionalnosti:

- Pokretanje igre i izbor nivoaa težine
- Otkrivanje i uparivanje kartica
- Brojanje pokušaja
- Specijalni režimi kao što su *Blind Mode* i *Hard Mode*
- Vremensko ograničenje (tajmer)
- Pobjeda i kraj igre
- Ponovno pokretanje igre

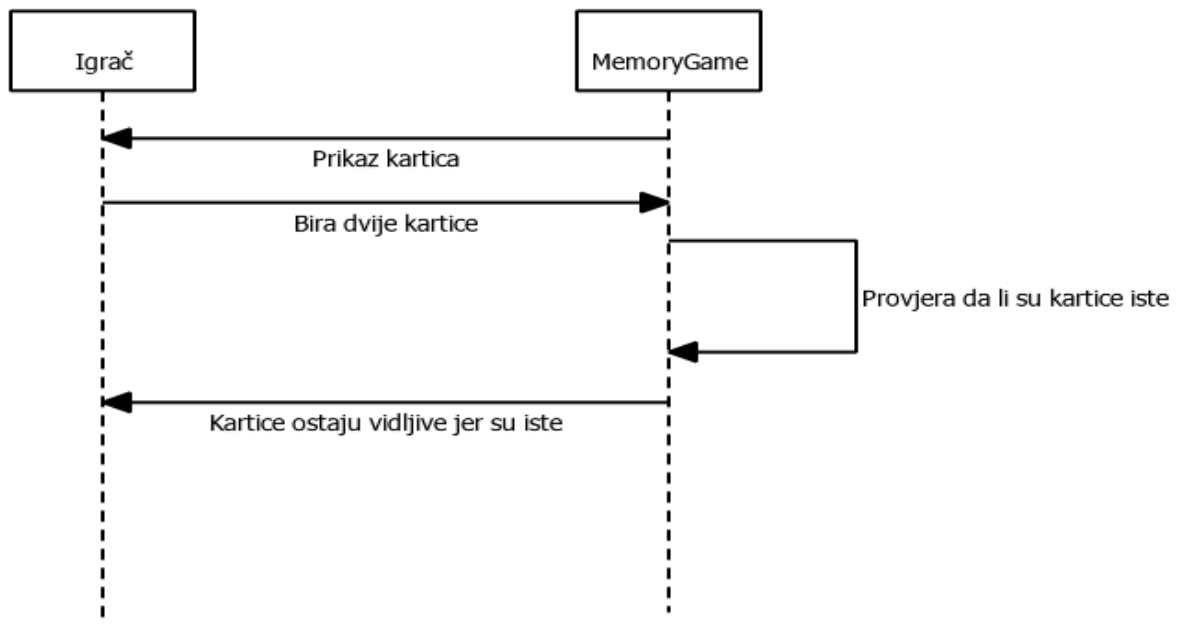
Svi ovi slučajevi upotrebe su izvedeni iz funkcionalnih zahtjeva sistema (REQ1 – REQ8) i predstavljaju osnovne tokove ponašanja aplikacije iz ugla krajnjeg korisnika.



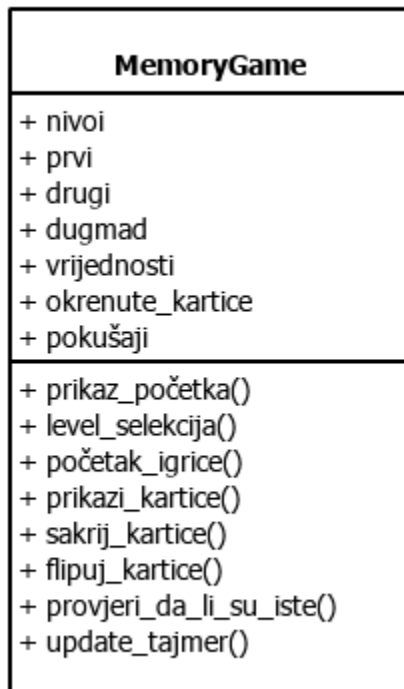
Dijagram sekvence za početak igrice



Dijagram sekvence za pogađanje karata

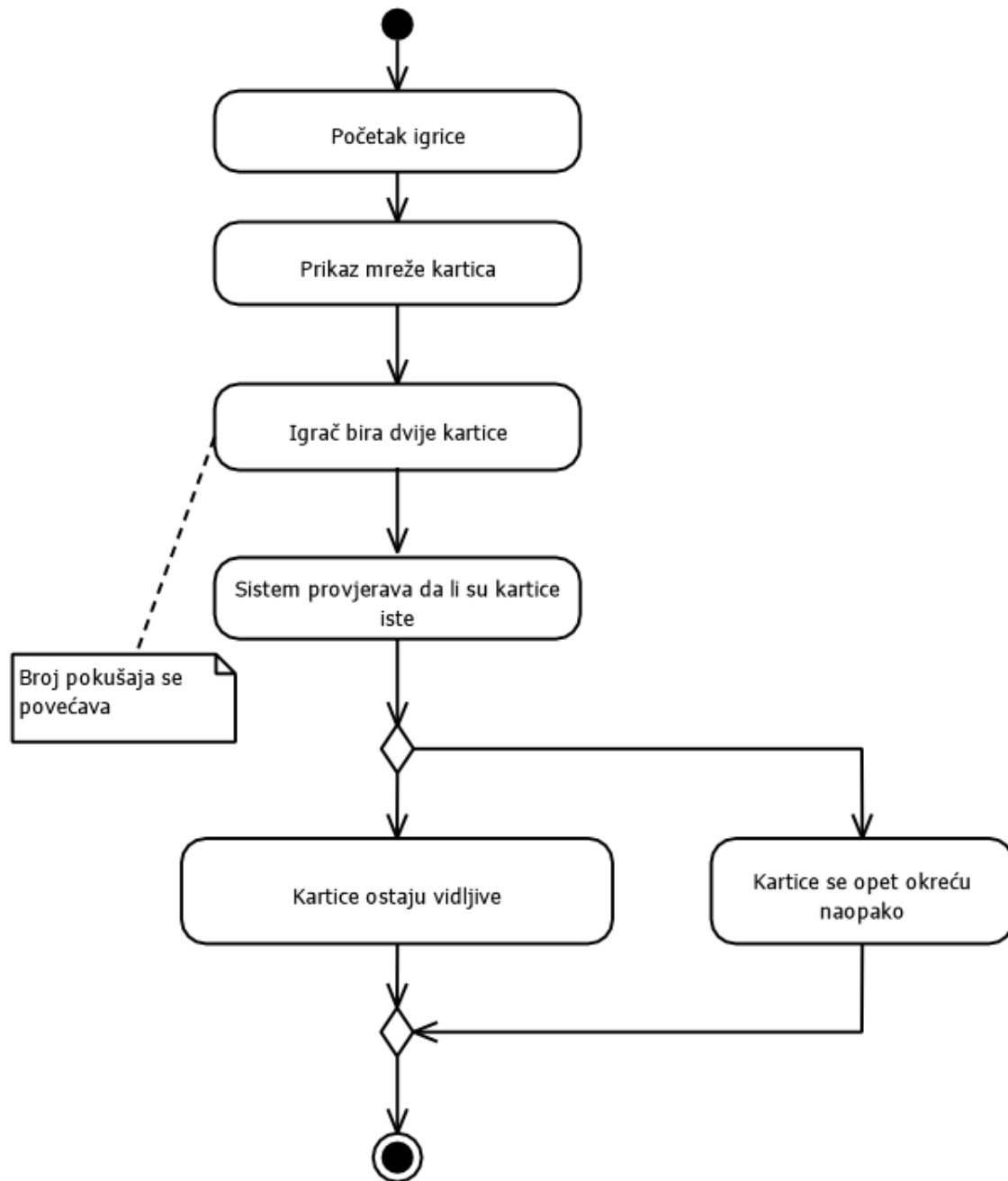


## Dijagram klase



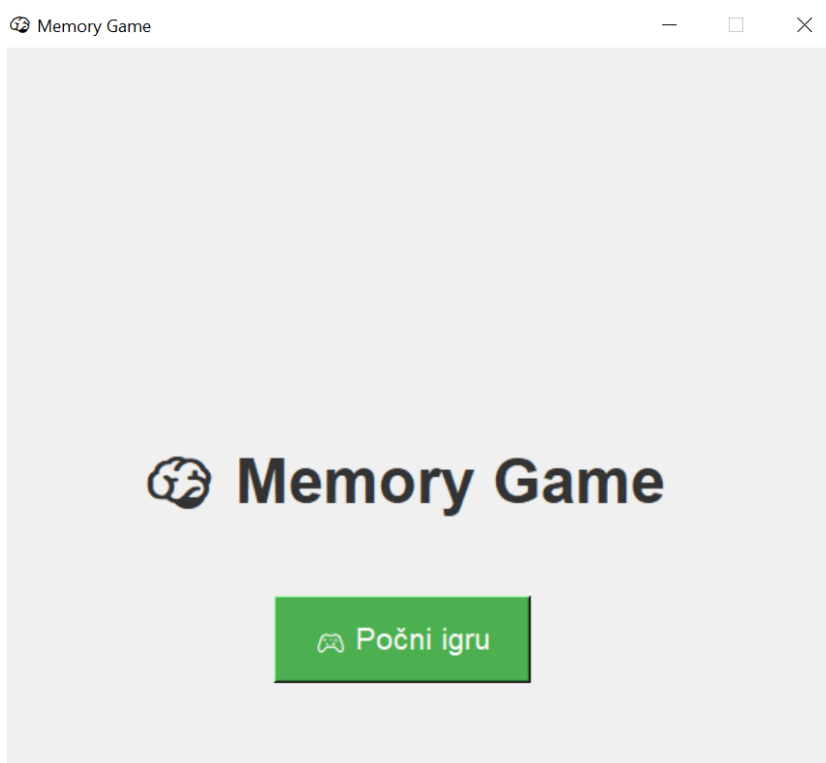
## Dijagram aktivnosti





## Interfejs


Početni ekran:




Odabir težine:





2x2

 Pokušaji: 0


 Nazad na izbor nivoa

4x4

 Pokušaji: 0


 Nazad na izbor nivoa

Blind mod

 Pokušaji: 0

8	7	3	4
8	6	1	2
2	3	5	4
7	5	6	1

 Nazad na izbor nivoa

Obavještenje



Vrijeme isteklo



Nažalost, isteklo je vrijeme.

OK



## Ključne funkcije i dijelovi koda

U konstruktorku klase postavljamo osnovne postavke prozora, definišemo nivoe igre i inicijalizujemo sve pomoćne varijable koje prate stanje igre (koliko je pokušaja bilo, koje kartice su već okrenute, mod igre i slično). Na kraju se poziva funkcija koja prikazuje početni ekran.

### Inicijalizacija klase

```
DEF __INIT__(SELF, ROOT):  
    SELF.ROOT = ROOT  
  
    SELF.ROOT.TITLE("🧠 IGRA PAMĆENJA")  
    SELF.ROOT.CONFIGURE(BG="#F0F0F0")  
    SELF.NIVOI = {  
        "🟢 LAKO (2X2)": 2,  
        "🟡 SREDNJE (4X4)": 4,  
        "🟣 TEŠKO (6X6)": 6,  
        "💧 HARD MODE (4X4 + TAJMER)": 4,  
        "👁️ BLIND MODE (4X4 MEMORIJA)}"  
    }  
    SELF.PRVI_IZABRANI = NONE  
    SELF.DRUGI_IZABRANI = NONE  
    SELF.DUGMAD = []  
    SELF.VRIJEDNOSTI = []  
    SELF.OKRENUTE = []  
    SELF.BROJ_POKUSAJA = 0  
    SELF.OZNAKA_TAJMERA = NONE  
    SELF.PREOSTALO_VRIJEME = 30  
    SELF.HARD_MODE = FALSE  
    SELF.BLIND_MODE = FALSE  
    SELF.PRIKAZI_POCETAK()
```

Funkcija prikazuje početni meni sa naslovom i dugmetom za početak igre. IKada igrač klikne na dugme, poziva se funkcija odabir\_nivoa koja omogućava izbor težine.

### Prikaz početnog ekrana

```
DEF PRIKAZI_PO CETAK(SELF):  
    SELF.OBRISI_SVE()  
  
    OKVIR = TK.FRAME(SELF.ROOT, BG="#F0F0F0")  
  
    OKVIR.PACK(EXPAND=TRUE)  
  
    NASLOV = TK.LABEL(OKVIR, TEXT="IGRA PAMĆENJA", FONT=("HELVETICA", 32, "BOLD"), FG="#333", BG="#F0F0F0")  
    NASLOV.PACK(PADY=30)  
  
    DUGME_PO CNI = TK.BUTTON(OKVIR, TEXT="POČNI IGRU", FONT=("HELVETICA", 16), BG="#4CAF50", FG="WHITE",  
                             ACTIVEBACKGROUND="#45A049", PADX=20, PADY=10, COMMAND=SELF.ODABIR_NIVOA)  
    DUGME_PO CNI.PACK(PADY=20)
```

Igrač bira jedan od nivoa igre. Klikom na neki nivo poziva se funkcija pocetak\_igre sa odgovarajućom veličinom mreže i imenom nivoa.

### Izbor nivoa težine

```
DEF ODABIR_NIVOA(SELF):  
    SELF.OBRISI_SVE()  
  
    SELF.HARD_MODE = FALSE  
    SELF.BLIND_MODE = FALSE  
  
    OKVIR = TK.FRAME(SELF.ROOT, BG="#F0F0F0")  
  
    OKVIR.PACK(EXPAND=TRUE)  
  
    LABELA = TK.LABEL(OKVIR, TEXT="ODABERI NIVO TEŽINE:", FONT=("HELVETICA", 20, "BOLD"), BG="#F0F0F0", FG="#333")  
    LABELA.PACK(PADY=20)  
  
    FOR NIVO, VELICINA IN SELF.NIVOI.ITEMS():  
        DUGME = TK.BUTTON(OKVIR, TEXT=NIVO, WIDTH=30, FONT=("HELVETICA", 14),  
                           BG="#2196F3", FG="WHITE", ACTIVEBACKGROUND="#1976D2",  
                           COMMAND=LAMBDA N=NIVO, V=VELICINA: SELF.POCETAK_IGRE(V, N))  
        DUGME.PACK(PADY=8)
```

## Pokretanje igre

```
DEF POCETAK_IGRE(SELF, VELICINA, IME_NIVOA):

    SELF.VELICINA = VELICINA

    SELF.UKUPAN_BROJ_KARTICA = VELICINA * VELICINA

    SELF.OBRISI_SVE()

    SELF.PRVI_IZABRANI = NONE

    SELF.DRUGI_IZABRANI = NONE

    SELF.DUGMAD = []

    SELF.VRIJEDNOSTI = []

    SELF.OKRENUTE = []

    SELF.BROJ_POKUSAJA = 0

    SELF.BROJ_UKLOPLJENIH = 0

    SELF.PREOSTALO_VRIJEME = 60

    SELF.HARD_MODE = "HARD MODE" IN IME_NIVOA

    SELF.BLIND_MODE = "BLIND MODE" IN IME_NIVOA

    BROJEVI = LIST(RANGE(1, SELF.UKUPAN_BROJ_KARTICA // 2 + 1)) * 2

    RANDOM.SHUFFLE(BROJEVI)

    SELF.VRIJEDNOSTI = BROJEVI

    SELF.OKVIR_INFO = TK.FRAME(SELF.ROOT, BG="#F0F0F0")

    SELF.OKVIR_INFO.PACK(PADY=10)

    SELF.OZNAKA_POKUSAJA = TK.LABEL(SELF.OKVIR_INFO, TEXT="🔄 POKUŠAJI: 0", FONT=("HELVETICA", 12), BG="#F0F0F0", FG="#555")

    SELF.OZNAKA_POKUSAJA.PACK(SIDE=TK.LEFT, PADX=10)

    IF SELF.HARD_MODE:

        SELF.OZNAKA_TAJMERA = TK.LABEL(SELF.OKVIR_INFO, TEXT=F"⌚ VRIJEME: {SELF.PREOSTALO_VRIJEME}S", FONT=("HELVETICA", 12),
        BG="#F0F0F0", FG="#C62828")

        SELF.OZNAKA_TAJMERA.PACK(SIDE=TK.RIGHT, PADX=10)

        SELF.AZURIRAJ_TAJMER()

    SELF.OKVIR_KARTE = TK.FRAME(SELF.ROOT, BG="#F0F0F0")

    SELF.OKVIR_KARTE.PACK(PADY=20)

    FOR I IN RANGE(SELF.UKUPAN_BROJ_KARTICA):

        DUGME = TK.BUTTON(SELF.OKVIR_KARTE, TEXT="", WIDTH=6, HEIGHT=3, FONT=("HELVETICA", 16, "BOLD"),

            RELIEF="RAISED", BG="#FFFFFF", FG="#333", BD=3,

            COMMAND=LAMBDA IDX=I: SELF.OKRENI_KARTICU(IDX))

        DUGME.GRID(ROW=I // VELICINA, COLUMN=I % VELICINA, PADX=6, PADY=6)

        SELF.DUGMAD.APPEND(DUGME)

    SELF.DUGME_NAZAD = TK.BUTTON(SELF.ROOT, TEXT="⬅ NAZAD NA IZBOR NIVOA", FONT=("HELVETICA", 12),

        BG="#E0E0E0", COMMAND=SELF.ODABIR_NIVOA)

    SELF.DUGME_NAZAD.PACK(PADY=10)

    IF SELF.HARD_MODE OR SELF.BLIND_MODE:

        SELF.PRIKAZI_SVE_KARTE()
```

Kada igrač klikne na karticu, ona se okreće tako što se prikaže broj koji ima ta kartica. Pamte se indeksi kliknutih kartica. Kada se klikne na drugu karticu, poziva se funkcija za provjeru poklapanja brojeva.

## Okretanje kartica

```
DEF OKRENI_KARTICU(SELF, IDX):  
    IF IDX IN SELF.OKRENUTE OR SELF.DRUGI_IZABRANI IS NOT NONE:  
        RETURN  
    IF SELF.BLIND_MODE:  
        IF SELF.PRVI_IZABRANI IS NONE:  
            SELF.PRVI_IZABRANI = IDX  
        ELIF SELF.PRVI_IZABRANI != IDX AND SELF.DRUGI_IZABRANI IS NONE:  
            SELF.DRUGI_IZABRANI = IDX  
            SELF.ROOT.AFTER(400, SELF.PROVJERI_POKLAPANJE)  
        RETURN  
    DUGME = SELF.DUGMAD[IDX]  
    DUGME.CONFIG(TEXT=STR(SELF.VRIJEDNOSTI[IDX]), STATE="DISABLED", DISABLEDFOREGROUND="BLACK",  
BG="#BBDEFB")  
    IF SELF.PRVI_IZABRANI IS NONE:  
        SELF.PRVI_IZABRANI = IDX  
    ELIF SELF.PRVI_IZABRANI != IDX AND SELF.DRUGI_IZABRANI IS NONE:  
        SELF.DRUGI_IZABRANI = IDX  
        SELF.ROOT.AFTER(400, SELF.PROVJERI_POKLAPANJE)
```

Ako su izabrane kartice iste, one ostaju okrenute i označene kao par. Ako su različite, one se ponovo okreću naopako. Kada se upare sve kartice, igra se završava.

## Provjera kartica

```
DEF PROVJERI_POKLAPANJE(SELF):  
    SELF.BROJ_POKUSAJA += 1  
  
    SELF.OZNAKA_POKUSAJA.CONFIG(TEXT=F"POKUŠAJI: {SELF.BROJ_POKUSAJA}")  
  
    VRIJEDNOST_PRVE = SELF.VRIJEDNOSTI[SELF.PRVI_IZABRANI]  
    VRIJEDNOST_DRUGE = SELF.VRIJEDNOSTI[SELF.DRUGI_IZABRANI]  
  
    IF VRIJEDNOST_PRVE == VRIJEDNOST_DRUGE:  
        SELF.OKRENUTE.EXTEND([SELF.PRVI_IZABRANI, SELF.DRUGI_IZABRANI])  
  
        SELF.BROJ_UKLOPLJENIH += 1  
  
        IF SELF.BLIND_MODE:  
            SELF.DUGMAD[SELF.PRVI_IZABRANI].CONFIG(TEXT=STR(VRIJEDNOST_PRVE), STATE="DISABLED", BG="#C8E6C9")  
            SELF.DUGMAD[SELF.DRUGI_IZABRANI].CONFIG(TEXT=STR(VRIJEDNOST_DRUGE), STATE="DISABLED", BG="#C8E6C9")  
  
        IF SELF.BROJ_UKLOPLJENIH == SELF.UKUPAN_BROJ_KARTICA // 2:  
            MESSAGEBOX.SHOWINFO("ČESTITAMO!", F"POBIJEDILI STE ZA {SELF.BROJ_POKUSAJA} POKUŠAJA!")  
            SELF.ODABIR_NIVOA()  
  
    ELSE:  
        IF NOT SELF.BLIND_MODE:  
            SELF.DUGMAD[SELF.PRVI_IZABRANI].CONFIG(TEXT=" ", STATE="NORMAL", BG="FFFFFF")  
            SELF.DUGMAD[SELF.DRUGI_IZABRANI].CONFIG(TEXT=" ", STATE="NORMAL", BG="FFFFFF")  
  
        IF SELF.BLIND_MODE AND VRIJEDNOST_PRVE != VRIJEDNOST_DRUGE:  
            PASS  
  
    SELF.PRVI_IZABRANI = NONE  
    SELF.DRUGI_IZABRANI = NONE
```

## Tajmer za hard mod

```
DEF AZURIRAJ_TAJMER(SELF):  
    IF NOT SELF.HARD_MODE:  
        RETURN  
  
    SELF.PREOSTALO_VRIJEME -= 1  
  
    IF SELF.OZNAKA_TAJMERA:  
        SELF.OZNAKA_TAJMERA.CONFIG(TEXT=F"🕒 VRIJEME: {SELF.PREOSTALO_VRIJEME}S")  
  
    IF SELF.PREOSTALO_VRIJEME == 0:  
        MESSAGEBOX.SHOWWARNING("🕒 VRIJEME ISTEKLO", "NAŽALOST, ISTEKLO JE VRIJEME.")  
  
        SELF.ODABIR_NIVOAA()  
  
    ELSE:  
        SELF.ROOT.AFTER(1000, SELF.AZURIRAJ_TAJMER)
```

Pregled literature, gotovih rješenja i softverske dokumentacije

<https://www.geeksforgeeks.org/python-gui-tkinter/>

<https://www.simplilearn.com/tutorials/python-tutorial/python-graphical-user-interface-gui>