

Simulator Logičkih Sklopova (SLS)

Tehnička dokumentacija



EXAMPLE

Autor: Luka Beg

Sadržaj tehničke dokumentacije

1. Uvod.....	5
1.1. O autoru	5
1.2. O ideji projekta.....	6
2. O logičkim sklopovima i SLS	7
2.1. Što je Simulator Logičkih Sklopova?.....	8
2.2. Originalnost ideje	8
2.3. Softverski alati korišteni.....	8
2.4. Načini rada programa SLS	9
3. Opis rada programa.....	10
3.1. Početni izbornik.....	10
3.2. Izbornik za odabir načina rada programa	11
3.3. Izbornik za stvaranje i učitavanje izazovnih razina.....	11
Čarobnjak za stvaranje razina	12
Čarobnjak za učitavanje razina	13
3.4. Izbornik s interaktivnim uputama za logičke sklopove	14
3.5. Izbornik s postavkama rada programa.....	15
3.5.1. Jačina tona.....	16
3.5.2. Kvaliteta prikaza	16
3.5.3. Rezolucija ekrana	16
3.5.4. Fullscreen način prikazivanja	16
3.5.5. VSync način prikazivanja	16
3.5.6. Veličina logičkih sklopova	16
3.5.8. Jezik korisničkog sučelja programa.....	17
3.5.9. Standard logičkih sklopova	17
3.5.10. Pozadina programa	17
3.5.11. Spremanje postavki.....	17
3.5.12. Zadane (tvorničke) postavke.....	17
3.6. Prozor „O projektu“	18
3.7. Izbornik tijekom pauziranja programa.....	18

3.8.	Pješčanik.....	18
3.9.	Izazovni način rada programa	19
3.10.	Napredni način rada programa	20
3.11.	Tablica istinitosti	20
3.12.	Povezivanje logičkih sklopova u SLS	21
3.13.	Rukovanje greškama	22
3.13.1.	Spajanje dvaju ilegalnih sklopova	22
3.13.2.	Spajanje ulaza i izlaza istog sklopa	22
3.13.3.	Tražena datoteka o spremljenim postavkama nije pronađena	23
3.14.	Lista skraćenica tipkovnice.....	23
4.	Tehnička Dokumentacija	24
4.1.	Lista značajki	24
4.1.1.	Postavke rada programa.....	24
4.1.2.	Korisnički izbornici u programu	24
4.1.3.	Pješčanik.....	24
4.1.4.	Izazovni način rada programa	25
4.1.6.	Općenite postavke o programu.....	25
4.1.7.	Ostalo.....	25
4.2.	Sistemska konfiguracija i specifikacije	26
4.2.1.	Minimalna sistemska konfiguracija računala	26
4.2.2.	Preporučena sistemska konfiguracija računala	26
4.2.3.	Potreban softver	26
4.2.4.	Optimizacija.....	26
4.3.	Napredne informacije o programu	27
4.3.1.	Unity podaci	27
4.3.2.	Kompilacijske postavke	27
4.3.3.	Prikazivanje zaslona – World Space način platna	27
4.3.4.	Reproduciranje zvučnih efekata.....	28
4.3.5.	Grafika i teksture	28
4.3.6.	Vlastiti namespace	28
4.4.	Program GameManager.cs	28
4.4.1.	Povezivanje sklopova	28
4.4.2.	Standard i jezik za tekst i grafiku	29

4.5.	Pokazivači u programu	29
4.6.	Sigurnost	30
4.7.	Proces izrade programa	30
4.7.1.	Ideja programa i programsko sučelje.....	30
4.7.2.	Prototip programa	30
4.7.3.	Prva verzija programa	30
4.7.4.	Druga verzija programa	31
4.7.5.	Treća verzija programa	31
4.7.6.	Održavanje programa.....	31
4.7.7.	Testiranje programa	31
4.8.	Unity Tagovi u programu.....	32
4.9.	Grafički prikaz rada programa	33
4.10.	Popis C# skripti u programu	36
4.11.	Jezici korisničkog sučelja u programu	40
4.12.	Računanje tablice istinitosti	40
5.	Budućnost Simulatora Logičkih Sklopova	41
6.	Zaključak	42
7.	Literatura.....	42
8.	Sadržaj slika u tehničkoj dokumentaciji	43

1. Uvod

1.1. O autoru

Moje ime je Luka Beg i učenik sam 1. razreda Tehničke škole u Karlovcu.

Imam 15 godina i rođen sam u gradu Karlovcu. Od ranijih dana zanima me rad s računalima i elektronikom.

Programiranjem se bavim već duže vrijeme. Prvi programski jezik koji sam naučio koristiti samostalno je bio **C#** koji se najčešće rabi u programskoj podršci **Unity**. Uz C#, ubrzo sam naučio te počeo koristiti **Python** i **C++**, dok **HTML** i **Java** tek sada učim.

Programsku podršku Unity koristim od početka programiranja, dakle od svoje 8 godine.

Školske godine 2022./2023. uspio sam prvi put proći na državnu razinu natjecanja iz informatike u kategoriji „**Osnove informatike**“.

Školske godine 2023./2024. ponovo sam pozvan na državnu razinu natjecanja iz informatike, ali ovog puta u obje kategorije: „**Osnove informatike**“ i „**Razvoj softvera**“.



Slika 1. – Luka Beg, Državno natjecanje iz Informatike 2023. godine



Slika 2. – Luka Beg, Državno natjecanje iz Informatike 2024. godine

1.2. O ideji projekta

Tijekom priprema za državno natjecanje 2023. godine, uočio sam da se mnogo zadataka u primjerima sastojalo od logičkih sklopova. U to vrijeme nisam ništa znao o logičkim sklopovima, jer ih još u školi nismo ni učili, tako da sam pomoć tražio po svuda, ali nisam ništa korisno uspio pronaći jer mnogo rješenja nije bilo prilagođeno početnicima. Nakon državnog natjecanja, saznao sam za kategoriju razvoja softvera, te sam shvatio da bi ideja o logičkim sklopovima bila savršena za to natjecanje. Na projektu radim već duže vrijeme, od početka prethodnih ljetnih školskih praznika (lipnja 2023. godine).



Slika 3. – Luka Beg, Državno natjecanje iz Informatike 2024. godine, prezentacija rada SLS



Slika 4. – Logotip programa SLS

2. O logičkim sklopovima i SLS

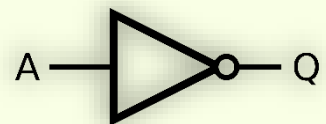
Logički sklopovi osnovne su gradivne jedinice svih procesora, mikročipova i integriranih krugova. Oni svoju primjenu nalaze svugdje oko nas, od našeg pametnog mobitela, do rakete za bilo koje nebesko tijelo. Zbog njihove primjene, vrlo su važni za razvoj tehnologije i modernizaciju.

Logički sklopovi služe za obradu digitalnih signala (koji mogu imati samo dvije vrijednosti: **1**, tj. uključen i **0**, tj. isključen signal).

Najčešće se koriste u elektrotehnici, elektronici, kemiji, mehanici, matematici, robotici i biologiji, a sastoje se od sljedećih osnovnih sklopova:

❖ **Sklop ne** (*NOT gate, inverter gate*):

- Vraća izlaz koji je suprotne vrijednosti od ulaza (npr. ako je 1. ulaz uključen, izlaz će biti isključen i obrnuto).
- Booleov izraz: \underline{A}



Slika 5. – Simbol za logički sklop NE (IEC/ANSI, NOT gate)

❖ **Sklop ili** (*OR gate*):

- Vraća uključen izlaz kada je bilo koji od 2 ulaza uključen (npr. ako je samo 1. ulaz uključen, izlaz će biti također uključen)
- Booleov izraz: $A + B$



Slika 6. – Simbol za logički sklop ILI (IEC/ANSI, OR gate)

❖ **Sklop i** (*AND gate*):

- Vraća uključen izlaz samo kad su oba ulaza uključena. Ako je samo jedan ulaz uključen, izlaz će ostati isključen. (npr. ako je su 1. i 2. ulazi uključeni, izlaz će biti uključen)
- Booleov izraz: $A \cdot B$



Slika 7. – Simbol za logički sklop i (IEC/ANSI, AND gate)

2.1. Što je Simulator Logičkih Sklopova?

Simulator Logičkih Sklopova™ (SLS™) obrazovni je program čiji je cilj poticati djecu i mladež u osnovnim i srednjim školama da uče o logičkim sklopovima i vježbaju logičko zaključivanje. Program je dizajniran na način da ga mogu koristiti sve vrste korisnika – od djece u osnovnim školama do studenata te čak inženjera.

SLS moguće je koristiti na svim mobilnim platformama poput **Android** i **iOS**, računalnim platformama poput **Windows**, **Linux** i **MacOS**, web platformama poput **WebGL** i konzolskim platformama poput **PlayStation**, **Xbox** i **Nintendo Switch**.

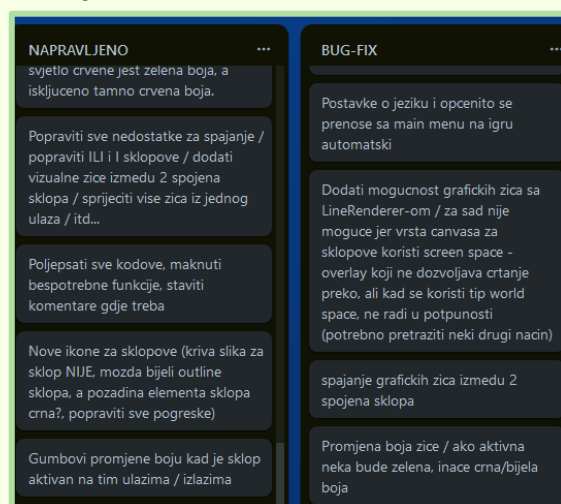
2.2. Originalnost ideje

Uz prisutnost drugih programa koji nude simulacije, tablice istinitosti i logičke sklopove, moj softver **SLS** jedinstven je u tome da je vrlo jednostavan za korištenje. Ne sadrži višak nepotrebnih postavki i opcija ili složene sklopove. Prilagođen je učenicima i studentima koji možda ne posjeduju predznanje o logičkim sklopovima ili o samome programu kako bi ga mogli slobodno koristiti, a veza s internetom nije uopće potrebna (*offline*) za one koji nemaju pristup internetu. Zato mislim da je moj softver vrlo jedinstven na tržištu.

2.3. Softverski alati korišteni

Alate koje sam koristio tijekom razvoja su:

- **Microsoft Visual Studio Community 2023¹**
- **Unity Editor² (verzija 2022.3.14f1)**
- **GIMP³**
- **Trello⁴**
- **Github⁵**



Slika 8. – Trello, web-stranica koju sam koristio tijekom razvoja softvera za razmjenjivanje ideja s mentorima te praćenja napretka

¹ [Microsoft Visual Studio](#) – uređivačko sučelje koda za programere (I.D.E.)

² [Unity Editor](#) – uređivačko sučelje korišteno za razvoj samog programa

³ [GIMP](#) – alternativni program **Photoshopa**, služi za stvaranje slika i grafika

⁴ [Trello](#) – web-stranica za praćenje napretka projekta, te raznih ideja

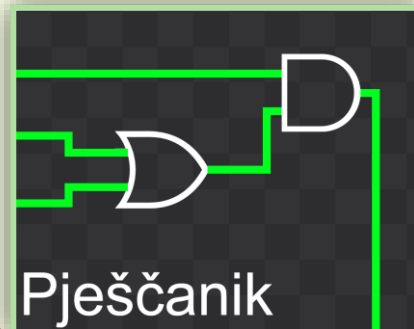
⁵ [GitHub](#) – web-stranica za praćenje napretka softvera, čuva sve inačice i verzije programa tijekom razvoja

2.4. Načini rada programa SLS

Koristeći **SLS**, korisnik može učiti, vježbati i razvijati logiku, kritičko razmišljanje i kreativnost vezane uz logičke sklopove. Postoji više načina rada programa:

Pješčanik:

Pješčanik dozvoljava slobodno spajanje i korištenje logičkih sklopova u svrhe zabave ili projektiranja bez ikakvih ograničenja ili bodovanja. Ovaj način rada programa prigodan je za zabavu i slobodno eksperimentiranje.



Slika 9. – Gumb u programu SLS za ulazak u pješčanik

Izazovni način rada programa:

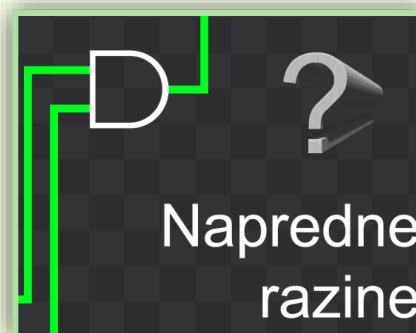
Izazovne razine dozvoljavaju korištenje razina sa vremenskim ograničenjima te ograničenjima odspajanja. Na ovakav zabavan način korisnik uči koristiti logičke sklopove i brzo logički zaključivati.



Slika 10. – Gumb u programu SLS za ulazak u izazovni način rada programa

Napredni način rada programa:

Napredne razine dozvoljavaju mirno korištenje izazovnih razina bez ikakvih vremenskih ograničenja ili ograničenja odspajanja. Također služi za eksperimentiranje, te učenje o logičkim sklopovima za one koji se ne vole natjecati.



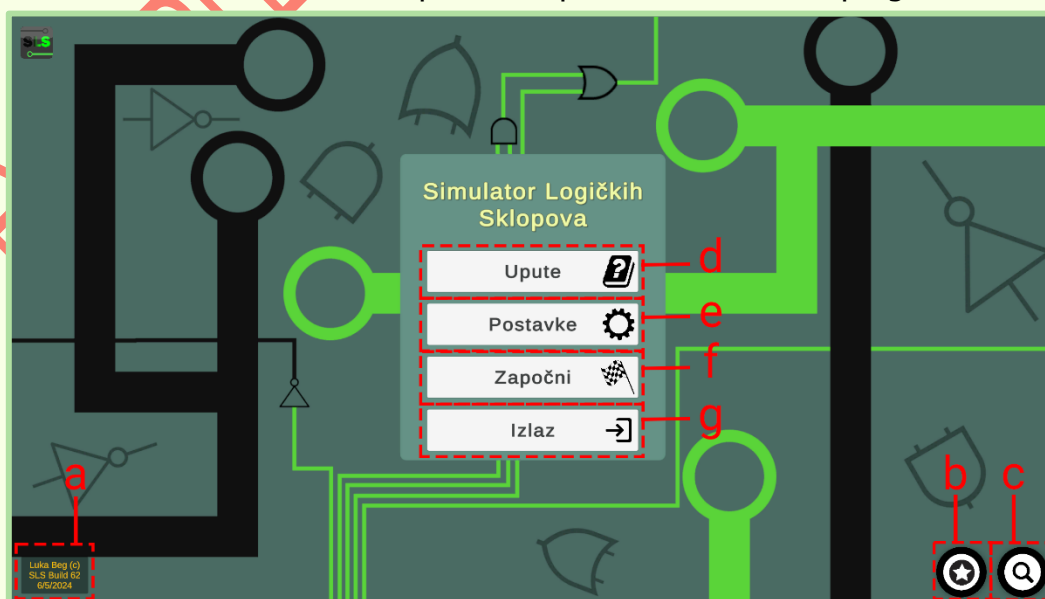
Slika 11. – Gumb u programu SLS za ulazak u napredni način rada programa

3. Opis rada programa

3.1. Početni izbornik

Pri ulasku u program, prikazat će se početni izbornik koji se sastoji od sljedećih opcija:

- ❖ **Verzija programa (a)**
 - Otvara prozor sa podacima o razvojnoj verziji programa
- ❖ **Ocjenjivanje programa (b)**
 - Otvara prozor u kojem se nalazi poveznica do upitnika za ocjenjivanje programa
- ❖ **O projektu (c)**
 - Otvara prozor sa podacima o samom projektu i autoru
- ❖ **Upute (d)**
 - Otvara prozor sa interaktivnim uputama za logičke sklopove
- ❖ **Postavke (e)**
 - Otvara prozor sa svim postavkama rada programa
- ❖ **Započni (f)**
 - Otvara prozor sa raznim načinima rada programa, te nakon izabranog načina prikazuje izbornik sa razinama
- ❖ **Izlaz (g)**
 - Otvara skočni prozor za potvrdu o izlasku iz programa.



Slika 12. – Početni izbornik programa SLS

3.2. Izbornik za odabir načina rada programa



Slika 13. – Izbornik za odabir načina rada programa SLS

Pritiskom na gumb za početak rada programa otvorit će izbornik za odabir načina rada programa.

Moguće je odabrati "Izazovne razine" (a), "Napredne razine" (b) i „Pješčanik“ (c). Prozor je moguće zatvoriti tipkom **Escape** ili pritiskom na gumb za zatvaranje prozora (d).

3.3. Izbornik za stvaranje i učitavanje izazovnih razina

Ulaskom u izazovne razine, otvoriti će se izbornik koji dozvoljava otvaranje čarobnjaka za stvaranje razina ili čarobnjaka za učitavanje razina.

Čarobnjak za stvaranje razina

Odabirom čarobnjaka za stvaranje razina otvoriti će se prozor sa svim postavkama razine. Korisnik može mijenjati vrijednosti strelicama **(b)** za dodavanje ili **(c)** za oduzimanje vrijednosti, te tipkovnicom.

Iz prozora moguće je izaći pritiskom na tipku **Escape**, ili pritiskom na gumb za zatvaranje prozora **(a)**.

Ako je količina ulaza **(d)** ili izlaza **(e)** manja od 1, pozadina polja postat će crvena, a tipka za spremanje **(g)** razine biti će onemogućena sve dok se vrijednost ne postavi na višu od 0, jer za rad razine potrebni su barem jedan ulaz i jedan izlaz.

Stvaranje razine

Količina sklopova I > 0 <

Količina sklopova ILI > 0 <

Količina sklopova NE > 0 <

Količina ULAZA > 0 <

Količina IZLAZA > 0 <

Vrijeme rješavanja (sekunde) > Unesite broj <
*ostavi prazno za neograničeno

Dozvoljen broj odspajanja > Unesite broj <
*ostavi prazno za neograničeno

Naziv datoteke Unesite naziv

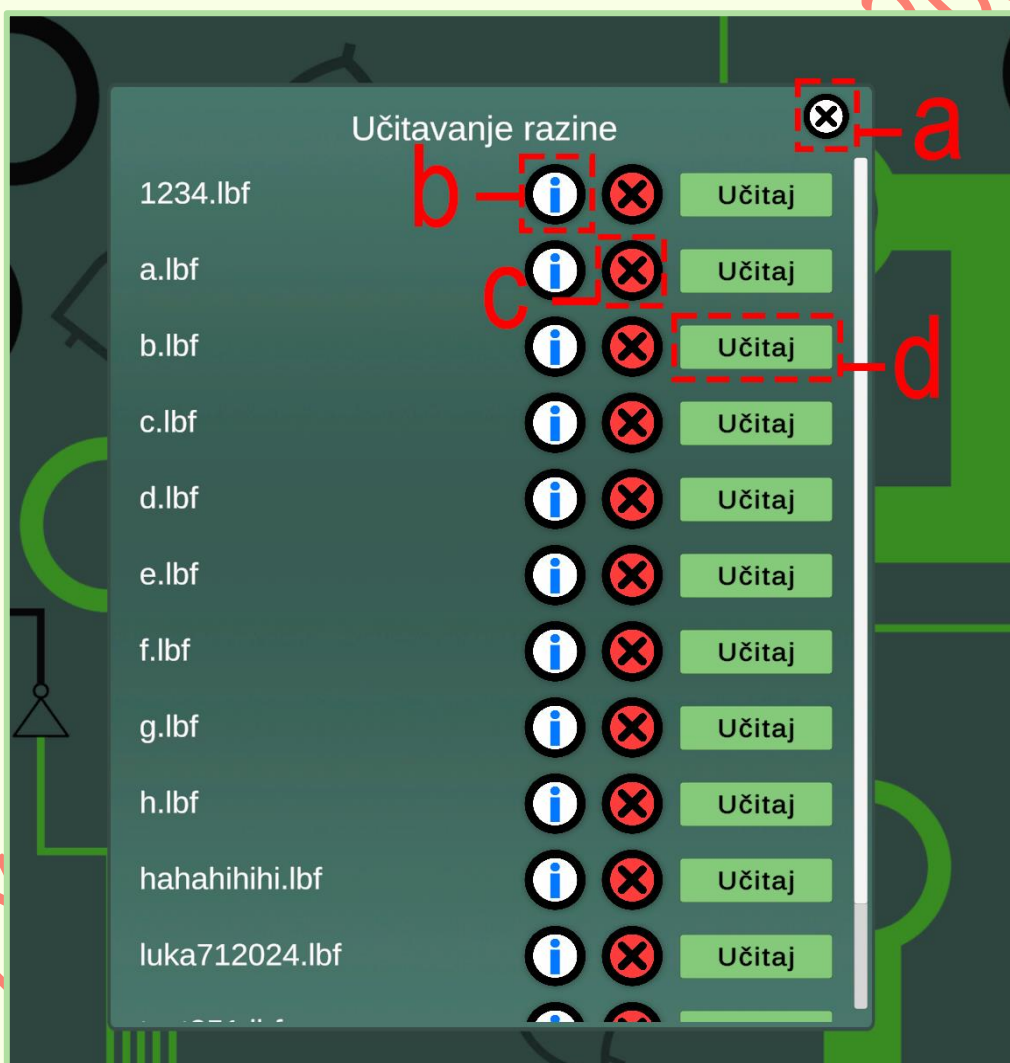
Spremi

Slika 14. – Čarobnjak za stvaranje razina programa SLS

Sva polja za unos dozvoljavaju unos **alfanumeričkih** vrijednosti (1,2,3,4,5,6,7,8,9,0), dok polje za unos naziva datoteke podržava samo brojeve i slova (bez simbola). Ako želimo, vrijeme rješavanja i dozvoljen broj odspajanja može biti **neograničen ukoliko se polje ostavi prazno**.

Čarobnjak za učitavanje razina

Odabirom na učitavanje razina, otvoriti će se prozor sa svim datotekama razina koje su pronađene na uređaju.



Slika 15. – Izbornik za učitavanje spremljenih razina programa SLS

Prozor za učitavanje razina moguće je zatvoriti pritiskom na tipku **Escape**, ili pritiskom na gumb za zatvaranje prozora **(a)**.

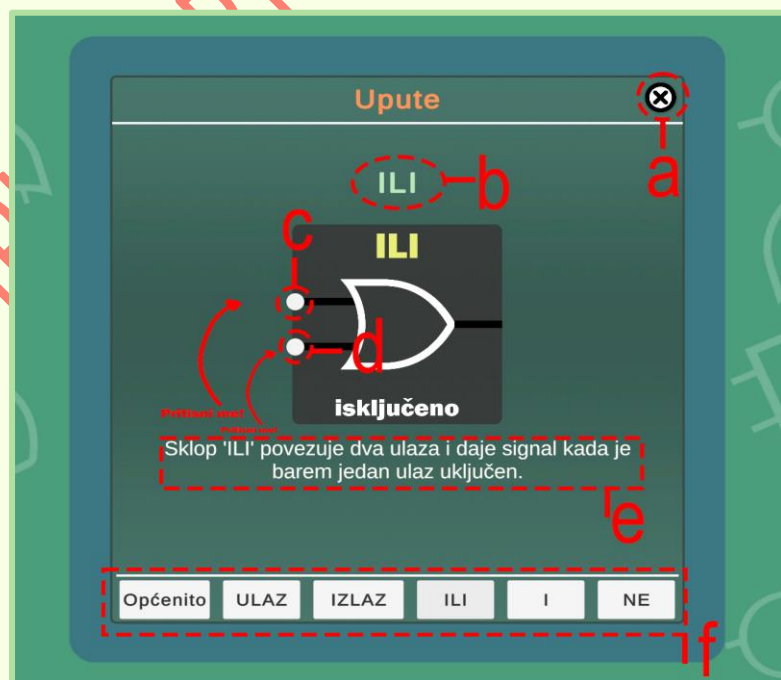
Sve prikazane datoteke moguće je izbrisati **(c)**, učitati **(d)** ili provjeriti **(b)**.

3.4. Izbornik s interaktivnim uputama za logičke sklopove

Pritiskom na opciju za interaktivne upute u početnom ili pauziranom izborniku otvorit će interaktivne upute za logičke sklopove. Svaki logički sklop u uputama sastoji se od:

- **Naslova sklopa (b)**
 - U naslovu nalazi se naziv sklopa
- **Interaktivne grafike sklopa**
 - Slika sklopa pomaže korisniku upamtiti sklop po simbolu i izgledu logičkih sklopova u programu. Svakom sklopu u uputama moguće je mijenjati stanja ulaza, te na taj način mogu razumjeti kako logički sklopovi reagiraju na željene uvjete (c) i (d)
- **Opisa sklopa (e)**
 - Opis sklopa je tekst koji opisuje sklop. U opisu se nalazi puni naziv sklopa te objašnjenje rada tog sklopa.

Na dnu prozora također se nalaze opcije kojima se dolazi do željenog sklopa (f), dok se na vrhu prozora nalazi gumb za izlazak na početni izbornik (a).



Slika 16. – Izbornik s interaktivnim uputama za logičke sklopove

3.5. Izbornik s postavkama rada programa

Pritiskom na gumb „Postavke” na početnom izborniku ili na pauziranom izborniku otvara se izbornik sa svim postavkama rada programa.

Iz izbornika s postavkama rada programa moguće je izaći pritiskom na tipku **Escape** na tipkovnici, ili pritiskom na tipku za izlaz u gornjem desnom kutu izbornika **(f)**.

Svaka postavka ima svoj naziv koji se mijenja ovisno o jeziku korisničkog sučelja koji je izabran **(b)**.

Postavke programa moguće je promijeniti na bilo koje željene opcije, **ali na kraju važno je spremi postavke pritiskom na gumb „Spremi” (e)**.



Slika 17. – Izbornik s postavkama rada programa SLS

Spremanjem postavki one će se pohraniti na uređaj i samostalno se učitati pri svakom pokretanju programa.

Pored svake postavke koja podržava mijenjanje pritiskom na funkcijske tipke na tipkovnici biti će prikazan kvadrat sa tom označenom F-tipkom **(a)**.
Izbornik s postavkama rada programa sastoji se od sljedećih postavki:

3.5.1. Jačina tona

Jačina tona jest klizač⁶ koji omogućuje promjenu jačine tona u programu. Pored postavke nalazi se tekst koji prikazuje vrijednost klizača **(c)**.

3.5.2. Kvaliteta prikaza

Kvaliteta prikaza također je padajući izbornik koji korisniku omogućuje odabir željene kvalitete prikaza grafike korisničkog sučelja. Postoje četiri opcije, a one su najviša, visoka, srednja i niska postavka kvalitete.

3.5.3. Rezolucija ekrana

Rezolucija ekrana padajući je izbornik⁷ koji omogućuje korisniku promjenu rezoluciju prikaza programa na ekranu. *Preporučeno je ne mijenjati ovu postavku!*

3.5.4. Fullscreen način prikazivanja

Fullscreen mode⁸ je opcija koja uključuje ili isključuje Fullscreen način prikazivanja prozora programa.

3.5.5. VSync način prikazivanja

VSyn⁹ je opcija koja uključuje ili isključuje VSync prikaz prozora programa.

3.5.6. Veličina logičkih sklopova

Veličina logičkih sklopova je klizač koji omogućuje korisniku promjenu veličine logičkih sklopova u programu. Ispod postavke za veličinu logičkih sklopova postoji i postavka za automatsku veličinu logičkih sklopova koja pri uključivanju zaključa klizač veličine, te automatski mijenja vrijednost veličine logičkih sklopova ovisno o količini logičkih sklopova u programu.

3.5.7. Veličina korisničkog sučelja

Veličina korisničkog sučelja je klizač koji omogućuje promjenu veličine korisničkog sučelja programa za lakšu navigaciju kroz sučelje.

⁶ **Klizač** – element korisničkog sučelja koji omogućava povlačenje kliznog elementa za postavljanje vrijednosti

⁷ **Padajući izbornik** – element korisničkog sučelja koji omogućuje odabir jedne od više ponuđenih opcija

⁸ **Fullscreen mode** – način prikazivanja prozora u kojem nije vidljiv obrub prozora, niti alatna traka OS

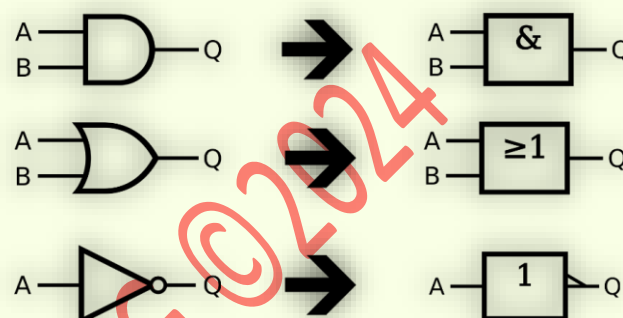
⁹ **Vsync** – način prikazivanja prozora u kojem se ne prikazuje višak sličica po sekundi (ako ekran može prikazivati najviše 60fps, onda će prozor programa biti ograničen na prikaz od 60fps)

3.5.8. Jezik korisničkog sučelja programa

Jezik korisničkog sučelja padajući je izbornik koji omogućuje korisniku odabir jezika korisničkog sučelja. Od ponuđenih opcija su Engleski jezik i Hrvatski jezik. Promjenom vrijednosti jezika korisničkog sučelja svi će se tekstovi unutar programa samostalno postaviti na izabran jezik.

3.5.9. Standard logičkih sklopova

Standard sklopova padajući je izbornik koji omogućuje korisniku promjenu standarda sklopova na **IEEE** ili **IEC/ANSI** standarde. Glavna razlika između **IEEE**¹⁰ i **IEC**¹¹/**ANSI** jest izgled simbola logičkih sklopova.



Slika 18. – Razlika između IEEE/ANSI (lijevo) i IEC (desno) simbola logičkih sklopova

3.5.10. Pozadina programa

Pozadina programa je padajući izbornik koji korisniku omogućuje odabir željene pozadine programa. U ponuđenim opcijama su tamna, svijetla, ljubičasta, zelena, crvena, narančasta i svjetlo-zelena pozadina.

3.5.11. Spremanje postavki

Pri svakom ulasku u program, učitava se datoteka „**settings.sls**” na putanji:

„[korisničko ime računala]\AppData\LocalLow\SLS\Simulator Logičkih Sklopova\settings.sls”

Ako datoteka **settings.sls** je pronađena na toj putanji, program će samostalno učitati prethodno postavljene postavke. Ako datoteka postavki nije pronađena, stvoriti će se nova datoteka postavki sa zadanim tvorničkim postavkama.

Pri spremanju postavki koristi se binarni sustav zapisivanja podataka. Vrsta datoteke **.sls** skraćeno je za „**Simulator Logičkih Sklopova**”.

3.5.12. Zadane (tvorničke) postavke

Ako je potrebno vratiti postavke na zadane, potrebno je samo pritisnuti na gumb „**Poništi postavke**” i vrijednosti postavka će se samostalno postaviti na zadane. **(d)**.

¹⁰ **IEEE** (Institute of Electrical and Electronics Engineers) – standard sklopova razvijen za države S.A.D.

¹¹ **IEC** (International Electrotechnical Commission) – standard sklopova razvijen za Europske države

3.6. Prozor „O projektu“

Pritiskom na gumb s ikonom povećala otvoriti će se izbornik u kojem je moguće pročitati sve o autoru, o ideji projekta, o problemima te o budućnosti programa.

3.7. Izbornik tijekom pauziranja programa

Zadnja tipka na gornjoj alatnoj traci u radu programa dozvoljava pauziranje rada programa, koje je također moguće učiniti pritiskom na tipku **Escape**. Od ponuđenih opcija pauziranog izbornika su:

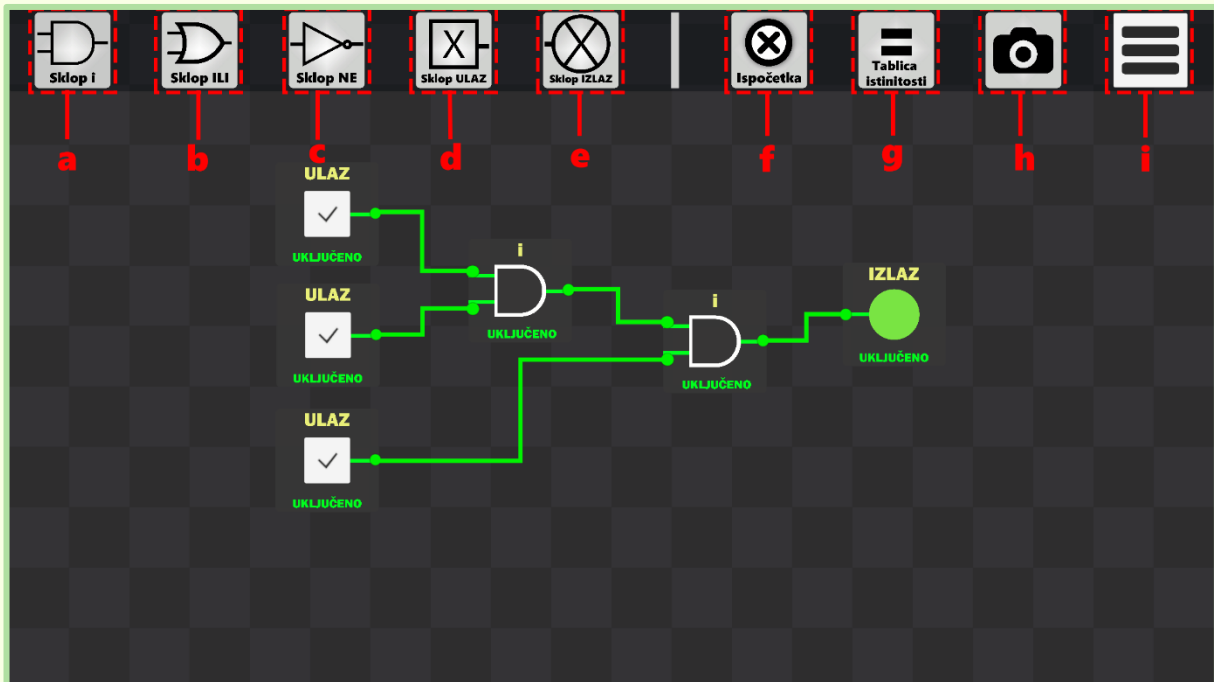
- **Nastavi**
 - Pritiskom na gumb zatvara se pauzirani izbornik i nastavlja rad
- **Upute**
 - Pritiskom na gumb otvara se izbornik s interaktivnim uputama o logičkim sklopovima
- **Postavke**
 - Pritiskom na gumb otvara se izbornik s postavkama programa
- **Početni izbornik**
 - Pritiskom na gumb zatvara se pauzirani izbornik i vraća na početni izbornik.
- **Izadi iz programa**
 - Pritiskom na gumb potpuno se izlazi iz programa na radnu površinu operativnog sustava

3.8. Pješčanik

Ulaskom u pješčanik, otvorit će se čista razina koja dozvoljava korištenje logičkih sklopova bez ikakvih ograničenja ili mjerenja.

Ovaj način programa služi za razvijanje kreativnosti, razmišljanja te projektiranje uz logičke sklopove.

Na gornjem dijelu ekrana nalazi se alatna traka koja omogućuje stvaranje sklopova (**a, b, c, d, e**), ponovno pokretanje razine (**f**), računanja tablice istinitosti (**g**), slikanje ekrana (**h**), te pauziranje programa (**i**).



Slika 19. – Pješčanik programa SLS

3.9. Izazovni način rada programa

Pritiskom na gumb „**Izazovne razine**“ na izborniku načina programa, otvorit će se način programa u kojoj korisnik može vježbati svoje razmišljanje, znanje i zaključivanje o logičkim sklopovima, uz brojač

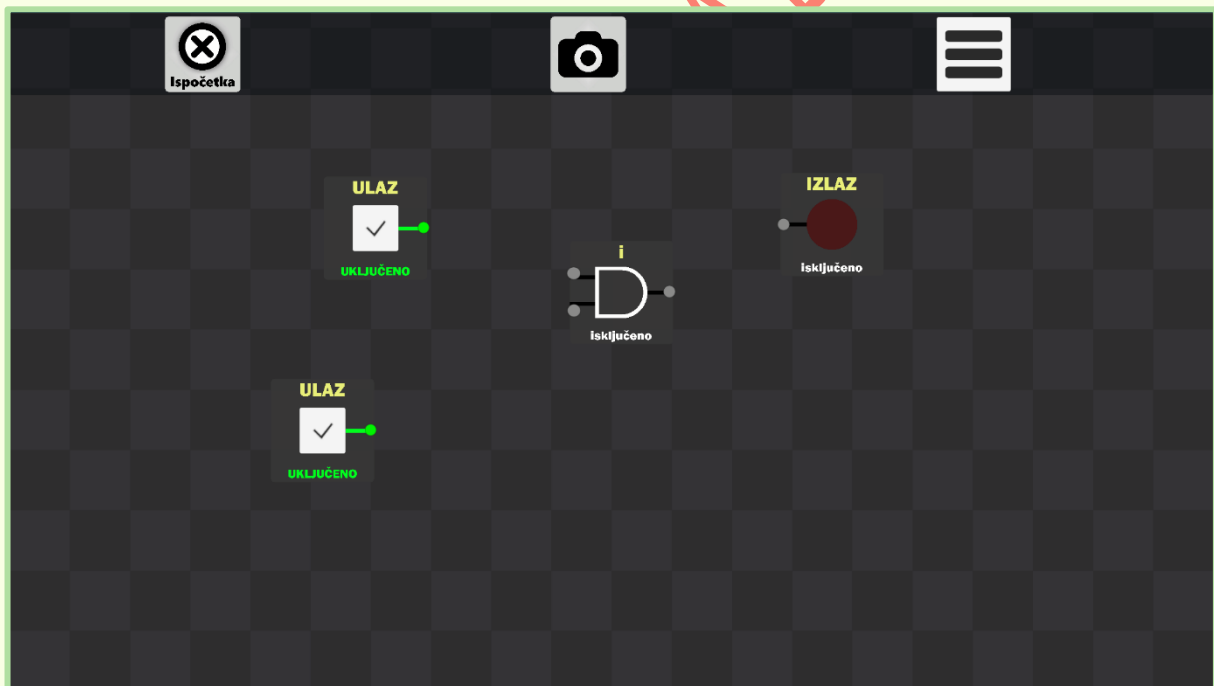


Slika 20. – Izazovni način rada programa SLS

vremena i brojač odspojenih sklopova tj. pogrešaka **(d)**, gumb za ponovo pokretanje razine **(a)**, gumb za slikanje zaslona **(b)** te gumb za otvaranje pauziranog izbornika **(c)**. Cilj ovog načina rada programa je da korisnik u što kraćem vremenu riješi razinu, tj. da iskoristi sve logičke sklopove i da su na kraju svi izlazi uključeni. Program pritom provjerava jesu li svi logički sklopovi iskorišteni kako ne bi došlo do pokušaja varanja.

3.10. Napredni način rada programa

Pritiskom na gumb „**Napredne razine**” na izborniku načina programa, otvorit će se način programa u kojoj korisnik može koristiti izazovne razine bez vremenskih ograničenja ili ograničenja odspajanja. Služi za mirno rješavanje razina. Cilj je također uključiti sve izlaze i pritom iskoristiti sve logičke sklopove.



Slika 21. – Napredni način rada programa SLS

3.11. Tablica istinitosti

U pješčaniku moguće je izračunati tablicu istinitosti. Prije računanja tablice istinitosti potrebno je imati barem jedan ulaz i samo je dozvoljen jedan izlaz. Korisnik može koristiti bilo koje logičke sklopove i neograničen broj ulaza za računanje tablice istinitosti.

Ulazi	Izlazi
000	0
001	0
010	0
011	0
100	0
101	0

Ukupno kombinacija: 8

Slika 22. – Tablica istinitosti u pješčaniku

Nakon što je tablica istinitosti u potpunosti izračunata, prikazat će se prozor sa svim rezultatima računanja.

Prozor za tablicu istinitosti moguće je zatvoriti pritiskom na tipku **Escape**, ili pritiskom na gumb za zatvaranje prozora **(a)**.

U stupcu s ulazima **(b)** nalaze se sve vrijednosti ulaza, dok se u stupcu s izlazima **(c)** nalaze sve vrijednosti izlaza (1 – Aktivan signal, 0 – neaktivan)

Pri dnu prozora nalazi se tekst **(d)** koji prikazuje broj mogućih kombinacija tablice istinitosti.

3.12. Povezivanje logičkih sklopova u SLS

Spajanje logičkih sklopova u programu SLS vrlo je jednostavno – potrebno je samo pritisnuti na željeni ulaz logičkog sklopa **(a ili b)**, te na željeni izlaz logičkog sklopa **(c)** (*redoslijed nije važan!*). Logički sklopovi također mogu se povlačiti slobodno po površini ekrana na način da se pritisne na obrub logičkog sklopa i povlači.



Slika 23. – Logički sklop ILI u programu SLS

3.13. Rukovanje greškama

Korisnik može često napraviti mnoge greške prilikom korištenja programa. To znači da može koristiti program u načinima za koje to nije prvotno zamišljeno. Zato je vrlo važno znati kako rukovati greškama u softveru.

3.13.1. Spajanje dvaju ilegalnih sklopova

Spajanje dvaju ilegalnih sklopova pogreška je koju mnogi početnici mogu napraviti. Pod „*spajanje dvaju ilegalnih sklopova*” misli se na spajanje dvaju sklopa koja nikako ne smiju biti spojeni, kao npr. spajanje **ulaza** i **ulaza**. To ne bi smjelo biti dopušteno, jer dva ulaza ne smiju ovisiti jedno o drugom.

Također nije dozvoljeno spojiti **izlaz** sa **izlazom**, jer oba sklopa su izlazna, te ne dovode nikakav signal, tako da ne bi imalo smisla dopustiti njihovo spajanje.

Ove pogreške vrlo je lako spriječiti u programu, jer možemo usporediti nazive oba sklopa koja se trebaju spojiti. Ako oba imena nisu dozvoljena, onda se briše spoj i nastavlja s radom programa.

3.13.2. Spajanje ulaza i izlaza istog sklopa

Spajanje ulaza i izlaza istog sklopa također nije dozvoljeno, jer nema smisla spojiti **ulaz sklopa NE** na **izlaz istog sklopa NE**. To bi prouzročilo beskonačnu petlju, koja bi vrlo brzo usporila cijeli uređaj i program bi prestao raditi.

Najlakši način za sprječavanje ove pogreške u programu je **provjera sličnosti sklopova koji se pokušavaju spojiti**. Npr. ako spojimo **izlaz sklopa NE** na **ulaz sklopa ILI**, program bi prvo provjerio ako su imena oba sklopa ista i ako imaju isti Unity tag. Ako nemaju sličnosti, onda bi nastavio sa procesom spajanja, ali ako bi imali slično ime i tag, onda bi izbrisali spoj i nastavili s radom programa.

3.13.3. Tražena datoteka o spremljenim postavkama nije pronađena

Svakim ulaskom u program samostalno se učitavaju spremljene postavke na uređaju. Pritom se traži datoteka „**settings.sls**“ na putanji:

„[korisničko ime računala]\AppData\LocalLow\SLS\Simulator
Logičkih Sklopova\settings.sls“

Ako traženi podatak o spremljenim postavkama nije pronađen, program će stvoriti datoteku **settings.sls** na toj putanji **sa zadanim postavkama** te ju učitati.

3.14. Lista skraćenica tipkovnice

U izborniku s postavkama rada programa, pored svake postavke koja podržava mijenjanje pritiskom na funkcijske tipke na tipkovnici biti će prikazan kvadrat sa tom označenom tipkom.

Program dozvoljava korištenje funkcijskih tipki za samo odabrane postavke

Tipka	Opis funkcije
ESC	Otvaranje pauziranog izbornika
F2	Promjena standarda sklopova (IEC i IEEE)
F3	Promjena jezika korisničkog sučelja (hrvatski i engleski)
F4	Promjena boje pozadina tijekom rada programa
F6	Prikaz statistike programa (FPS , u razvojne svrhe)
F9	Ponovo pokretanje cijele razine
F11	Slikanje trenutnog zaslona
F12	Promjena Fullscreen načina prozora

4. Tehnička Dokumentacija

4.1. Lista značajki

4.1.1. Postavke rada programa

- ✓ Odabir jezika korisničkog sučelja (**Engleski, Hrvatski**)
- ✓ Odabir standarda sklopova (**IEEE, IEC**)
- ✓ Odabir rezolucije ekrana (**1920x1080, 2560x1440, ...**)
 - Moguće je izabrati **refresh rate (Hz)**
 - **Program podržava sve rezolucije**, jer se veličina korisničkog sučelja prilagođava ovisno o izabranoj rezoluciji.
- ✓ Odabir kvalitete prikaza (**najviša, visoka, srednja, niska kvaliteta**)
- ✓ Odabir načina prikazivanja prozora (**Fullscreen, VSync**)
- ✓ Odabir jačine tona
- ✓ Odabir **veličine sklopova**
- ✓ Odabir pozadine (**tamna, svjetla, zelena, crvena, ljubičasta, svjetlo-zelena, narančasta**)
- ✓ Spremanje postavki na uređaj
- ✓ Samostalno učitavanje spremljenih postavki pri ulasku
- ✓ Mogućnost vraćanja na tvorničke (zadane) postavke

4.1.2. Korisnički izbornici u programu

- ✓ Početni izbornik
- ✓ Izbornik s postavkama rada programa
- ✓ Izbornik pauziranja rada programa
- ✓ Prozor s interaktivnim uputama o sklopovima
- ✓ Prozor s informacijama o projektu

4.1.3. Pješčanik

- ✓ Gornja alatna traka za stvaranje sklopova
- ✓ Računanje tablica istinitosti

4.1.4. Izazovni način rada programa

- ✓ Brojač vremena
- ✓ Brojač obrisanih spojeva logičkih sklopova

4.1.5. Napredni način rada programa

- ✓ Korištenje izazovnih razina bez ograničenja (bez brojača odspajanja i vremena)

4.1.6. Općenite postavke o programu

- ✓ Sklopovi: **I (AND), ILI (OR), NE (NOT), ULAZ (INPUT), IZLAZ (OUTPUT)**
- ✓ Spajanje sklopova s pritiskom na gumb koji se nalazi na ulazu i izlazu sklopa
- ✓ **Drag & Drop** sistem za povlačenje sklopova
- ✓ Žice koje se samostalno pomiču ovisno o poziciji logičkog sklopa
 - Žice također mijenjaju boju ovisno o stanju ulaza i izlaza sklopa koje povezuju
 - Ako kroz žicu prolazi aktivan signal, boja žice će biti zelena, u suprotnome će biti siva.
- ✓ Potpuno prilagođen sistem audio reprodukcije – razni zvukovi za spajanje, povlačenje, spuštanje sklopova, te za gumbе i postavke na izbornicima.

4.1.7. Ostalo

- ✓ Projekt je građen na Unity 2D verziji **URP** (*Universal Render Pipeline*) koristeći **IL2CPP** konverziju
- ✓ Internetska veza nije potrebna prilikom korištenja
- ✓ **Mogućnost kontaktiranja podrške** (sls.podrska@gmail.com)
- ✓ Slikanje trenutnog zaslona pritiskom **F11** tipke ili pritiskom na tipku sa ikonom kamere na gornjoj alatnoj traci

4.2. Sistemska konfiguracija i specifikacije

SLS optimiziran je za korištenje u većini uređaja. Potrebno je vrlo malo resursa za njegovo pokretanje i rad.

4.2.1. Minimalna sistemska konfiguracija računala

CPU	Bilo koji x86 ili x64 bitni procesor sa 2.0GHz
GPU	NVIDIA GT 1030 ili AMD Radeon RX 550
RAM	2GB RAM
DISK	~500 MB
OS	Windows 7

4.2.2. Preporučena sistemska konfiguracija računala

CPU	Intel I5 (2.5GHz) ili bolje
GPU	NVIDIA GT 1050, AMD Radeon RX 560 ili bolje
RAM	8GB RAM ili bolje
DISK	~1 GB
OS	Windows 10

4.2.3. Potreban softver

Za pokretanje i korištenje ovog programa nije potreban nikakav dodatan softver osim samog programa. Potrebno je samo imati operativni sustav **Windows**, **Linux** ili **MacOS** na računalima. **Android** ili **iOS** na mobitelima.

Internetska veza nije potrebna prilikom korištenja, već je samo potrebna prilikom preuzimanja čarobnjaka za instalaciju programa.

4.2.4. Optimizacija

Optimizacija softvera vrlo je važan proces tijekom razvoja softvera, jer je potrebno osigurati da će program raditi na svim platformama i sistemskim konfiguracijama. Zato je važno što bolje osmisлити program te paziti na urednost koda.

Tijekom razvoja ovog projekta, uložio sam dodatan trud i pazio da mnoge funkcije i dijelove koda napišem što bolje moguće.

4.3. Napredne informacije o programu

4.3.1. Unity podaci

Program je izgrađen na Unity verziji **2022.3.14f1**, a za ovaj projekt korišten je Universal Render Pipeline (**URP**). Universal Render Pipeline omogućava upotrebu projekta na raznim uređajima, uključujući pametne telefone (**Android** i **iOS**), igraće konzole (**PlayStation**, **Xbox** i **Nintendo Switch**), te operativne sustave (**Windows**, **Linux** i **MacOS**). Također je moguće koristiti URP programe na web-stranicama putem **WebGL-a**. URP podržava i uređaje s minimalnim sistemskim konfiguracijama poput školskih tableta.

Od **7. svibnja 2024.** godine, potpuni projekt se sastoji od **9102 linija vlastitog C# koda**.

4.3.2. Kompilacijske postavke

Tijekom izgradnje verzije programa, koristio sam **IL2CPP** metodu kompilacije (***I**ntermediate **L**anguage [**C#**] to **CPP** [**C++**]*). Ova metoda pretvara **C#** programski jezik u **C++**. Ovo je izuzetno korisno jer omogućava brže učitavanje programa, s obzirom na to da C++ efikasno prolazi kroz računalo. Također, otežava pristup drugima koji bi pokušali otvarati ili izdvajati podatke programa za njihove vlastite svrhe.

4.3.3. Prikazivanje zaslona – **World Space** način platna

Zbog potreba integracije crtanja žica između logičkih sklopova, bilo je potrebno koristiti **World Space** način 2D platna u Unity-u. To znači da se platno, tj. korisničko sučelje ne nalazi samo ispred kamere, ili u ovom slučaju ekrana, već u pravim koordinatama u sceni. Bilo je vrlo teško integrirati sustav crtanja crta, jer one ne mogu biti crtane preko korisničkog sučelja u normalnom načinu prikazivanja platna/ekrana, već je bilo potrebno prebaciti cijeli projekt na **World Space**.

4.3.4. Reproduciranje zvučnih efekata

Zvučni podaci u ovome programu nalaze se samo u **MP3** ili **WAV** formatu. Prilikom reproduciranja svakog zvučnog efekta, u scenu se dodaje novi izvor zvuka, koji se briše nakon što je zvuk u potpunosti reproduciran (dopušta se vremenski razmak od **0.5s** kako bi u potpunosti bio reproduciran a ne prekinut tijekom reproduciranja)

4.3.5. Grafika i teksture

U ovom programu većina grafike i tekstura nalaze se u **PNG** formatu.

4.3.6. Vlastiti namespace

SLS sadrži vlastiti namespace – **Luka**. Luka sadrži sve **enum** vrijednosti koje su potrebne za rad programa, poput **GateType**, **GameManagerType**, **CursorType** i **CurrentPlatform**.

4.4. Program GameManager.cs

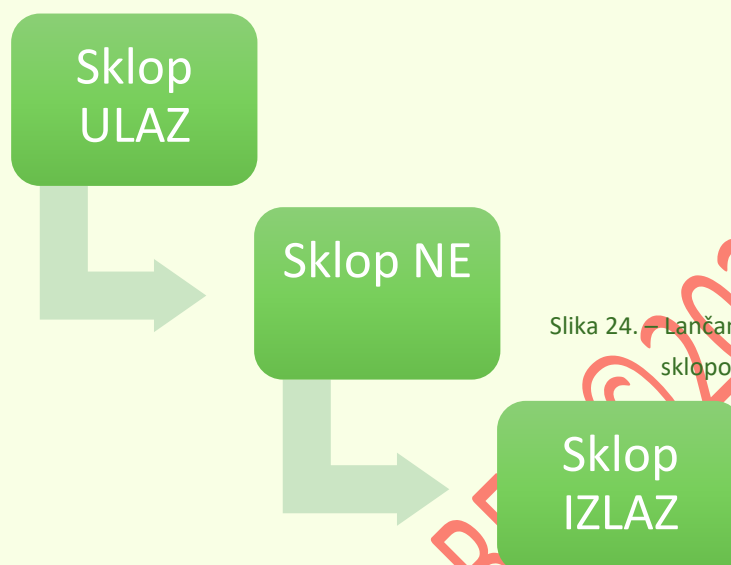
Za realizaciju funkcija koje su potrebne tijekom rada programa, napravio sam skriptu pod nazivom „**GameManager.cs**“. U njoj se nalazi mnogo korisnih dijelova koda koji su vrlo važni za opći rad programa.

4.4.1. Povezivanje sklopova

Povezivanje sklopova uspio sam riješiti na način da svaki sklop ima svoju skriptu koja kontrolira njegovu grafiku i logiku. Postoje skripte za sklop **NE**, sklop **ILI**, sklop **I**, **ULAZ** te **IZLAZ**, a svaki od njih sadrži vrijednosti ulaza i izlaza u varijabli tipa **boolean** (True i False).

Također, svaka skripta koja kontrolira logiku sklopa sadrži varijablu koja će pri svakom povezivanju spremi sklop s kojim je u vezi.

Glavna logika za spajanje sklopova nalazi se u **GameManager** skripti, u funkciji **Connect(sklopA, sklopB, jesamLiNaUlazuA)** koja spaja 2 objekta logičkih sklopova koja su navedena – **sklopA** i **sklopB**.



Slika 24. – Lančani prikaz spajanja logičkih sklopova u programu

4.4.2. Standard i jezik za tekst i grafiku

LanguageManager sadrži funkcije koje traže dvije slike ili dva teksta, te ovisno o odabranom jeziku ili standardu logičkih sklopova, vraćaju točnu sliku ili tekst. Svaki izbornik (**početni**, **pauzirani**, **postavke**, **o projektu**) ima svoju skriptu koja provjerava ako se jezik i standard sklopova promijenio.

4.5. Pokazivači u programu

Ikona	Naziv	Opis
	CURSOR_IDLE Luka.CursorType	Pokazivač koji je vidljiv pri mirovanju pokazivača preko ne interaktivnog dijela korisničkog sučelja u programu.
	CURSOR_HOVER Luka.CursorType	Pokazivač koji je vidljiv pri povlačenju pokazivača preko interaktivnih dijelova korisničkog sučelja (gumb, polje za unos) ili logičkih sklopova
	CURSOR_DRAG Luka.CursorType	Pokazivač koji je vidljiv pri povlačenju logičkih sklopova po površini ekrana

4.6. Sigurnost

SLS izoliran je program, koji nema nikakvih izravnih veza sa internetom. Ona je potrebna samo tijekom ažuriranja, ili preuzimanja samog programa, stoga mislim da je moj softver vrlo siguran.

4.7. Proces izrade programa

4.7.1. Ideja programa i programsko sučelje

Svaki dobar softver i projekt kreće s vrlo dobro razrađenom i proučenom idejom. Nakon što sam odlučio da ću krenuti raditi na simulatoru logičkih sklopova, trebao sam izabrati programsko sučelje za realizaciju.

Izabrao sam programsko sučelje **Unity**, jer u njemu već radim skoro 6 godina pa se potpuno mogu snaći u njemu.

4.7.2. Prototip programa

Prva verzija koju sam ikad napravio bila je prototipna verzija, u kojoj se nalazio vrlo jednostavan početni izbornik, te opcije koje su se koristile za ulaz i izlaz iz programa. U programu su se nalazili jednostavni logički sklopovi, koji su imali Drag & Drop¹² mogućnosti.

Prototipnu verziju podijelio sam s roditeljima i mentorima, te su mi dali puno ideja i pomogli sa nastavkom razvoja.

Prva prototipna verzija izašla je **u srpnju 2023. godine.**

4.7.3. Prva verzija programa

Nakon dva mjeseca dugog rada, uspio sam dovršiti prvu verziju, u kojoj se nalazio pješčanik, postavke, poboljšani početni izbornik, pauzirani izbornik, te logički sklopovi koji su mijenjali boje i mogli se spajati jedni s drugima. Trebalo mi je puno truda

¹² **Drag & Drop** – „povuci i spusti“, mogućnost spuštanja i povlačenja elementa sa klikom i pomicanjem računalnog miša

da dođem do prve verzije, ali sam bio vrlo zadovoljan kad sam ju završio.

Prva verzija izašla je **20. kolovoza 2023.** godine i podijelio sam ju s mentorima i roditeljima. Bili su vrlo zadovoljni s prvom verzijom, ali htio sam još poboljšati korisničko sučelje i dodati mogućnosti za korisnike.

4.7.4. Druga verzija programa

Druga verzija izašla je par dana poslije prve verzije, te je sadržavala puno više opcija na početnom izborniku, a bila je puno stabilnija i optimizirana. Imala je bolji početni izbornik i novu grafiku za logičke sklopove.

Dodao sam i opciju da se mogu promijeniti standardi logičkih sklopova te jezici korisničkog sučelja na Engleski i Hrvatski jezik.

4.7.5. Treća verzija programa

Treća, završna verzija programa izašla je u **veljači 2024. godine.** Ona je bila zadnja verzija koja je službeno izašla za Natjecanje iz Informatike u kategoriji „**Razvoj Softvera**“ 2024.

Dodao sam puno mogućnosti, promijenio sam izgled mnogih izbornika, te stvorio čarobnjak za instalaciju kako bi korisnici mogli što lakše preuzeti i instalirati program.

4.7.6. Održavanje programa

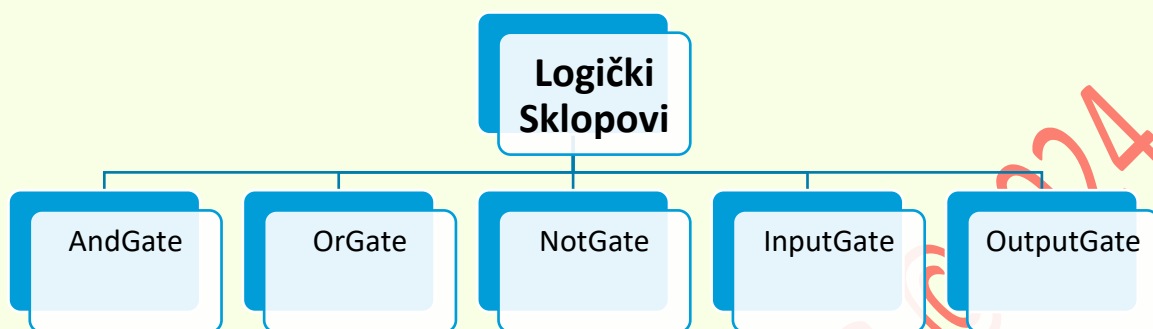
Redovito održavam program, dodavajući nove mogućnosti, popravke i optimizacije. Redovito prolazim kroz program tražeći nedostatke i saniram ih na način da neke stvari pojednostavim.

4.7.7. Testiranje programa

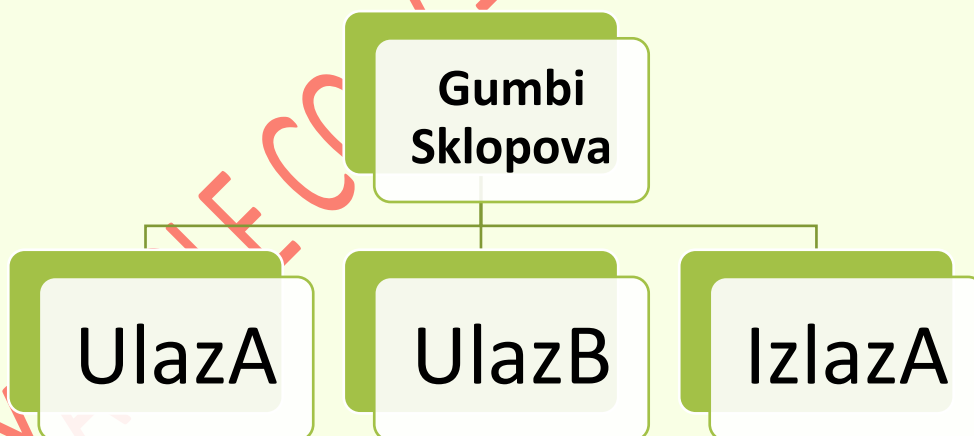
Testiranje programa proveo sam s prijateljima i obitelji.

Ustanovio sam kako je SLS pažljivo realiziran projekt, koji je vrlo učinkovit, dokumentiran i poučan program.

4.8. Unity Tagovi u programu



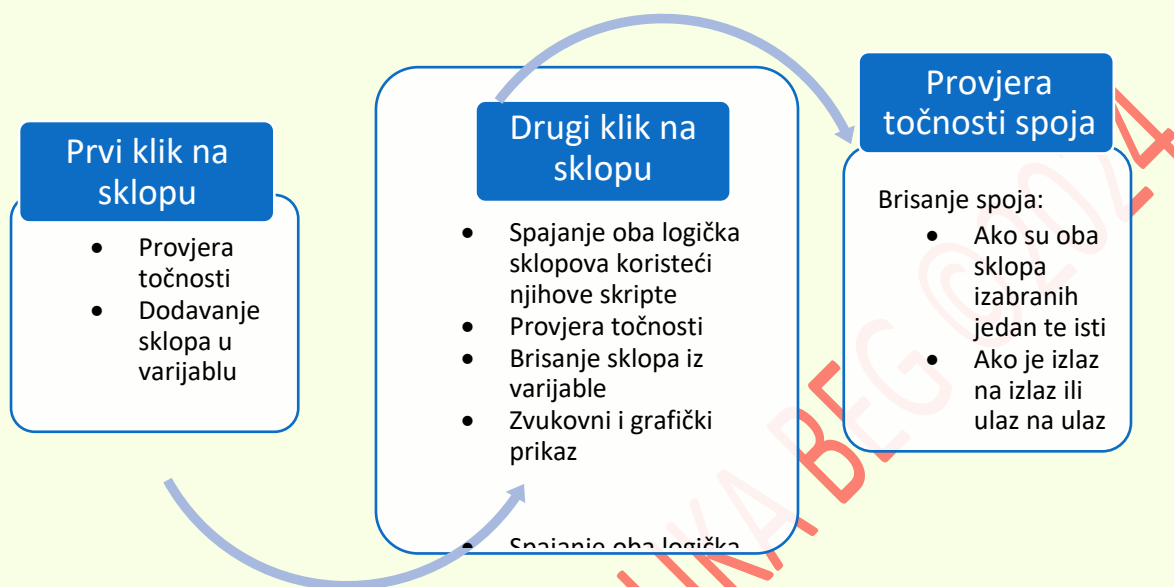
Slika 25. – Hijerarhija Unity tagova za razlikovanje vrsta logičkih sklopova u programu SLS



Slika 26. – Hijerarhija Unity tagova za razlikovanje ulaza (A i B) te izlaza logičkih sklopova u programu SLS

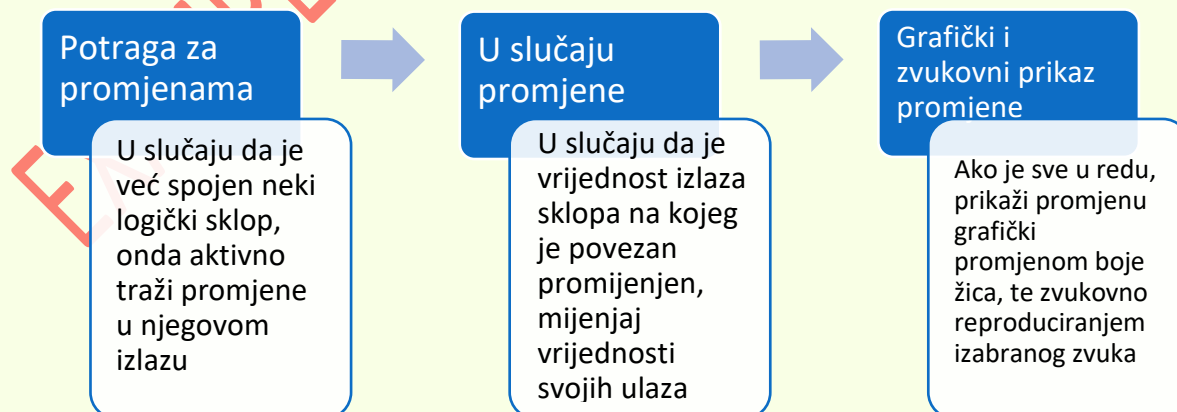
4.9. Grafički prikaz rada programa

Povezivanje sklopova

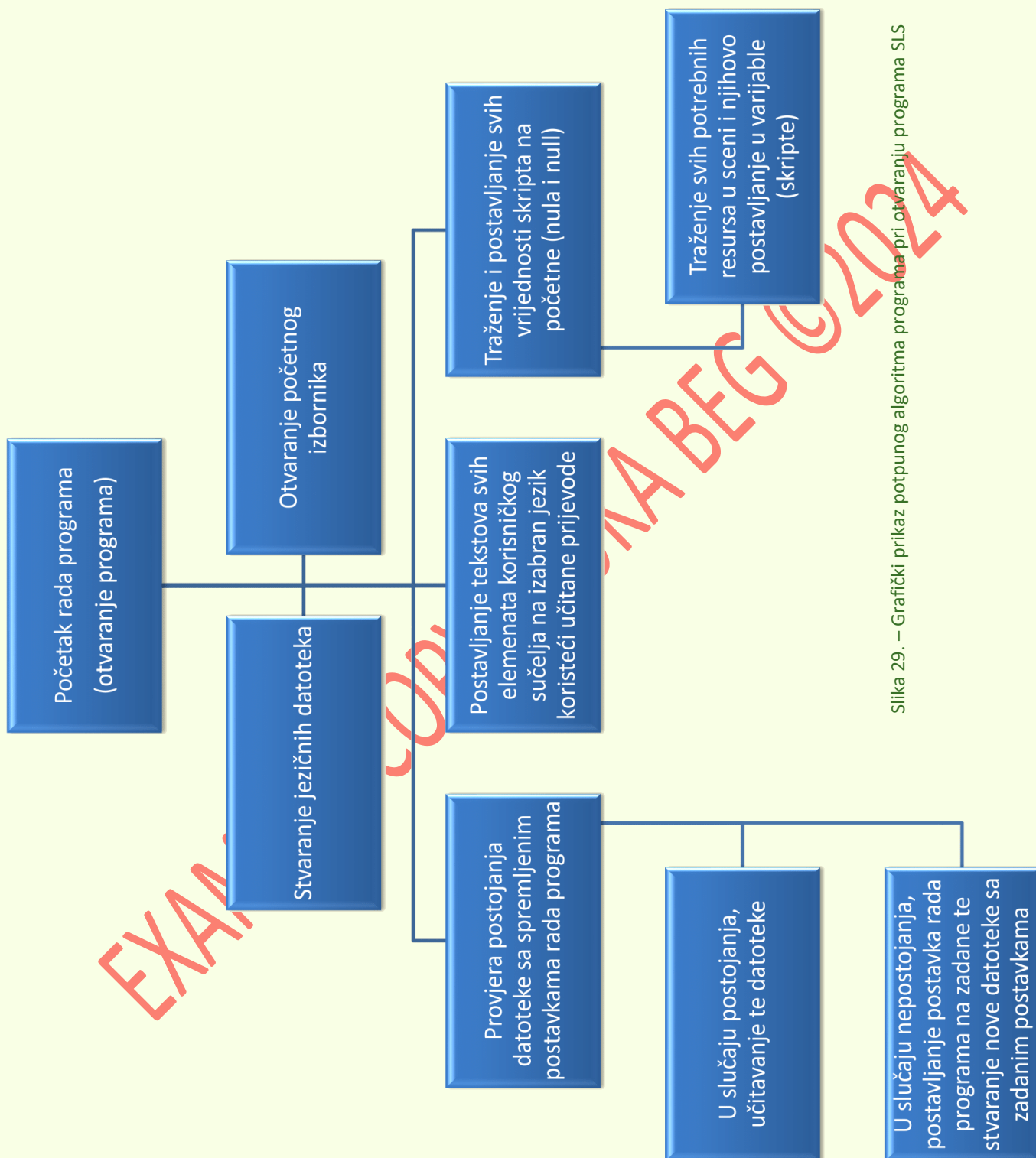


Slika 27. – Grafički prikaz algoritma za povezivanje logičkih sklopova u programu SLS

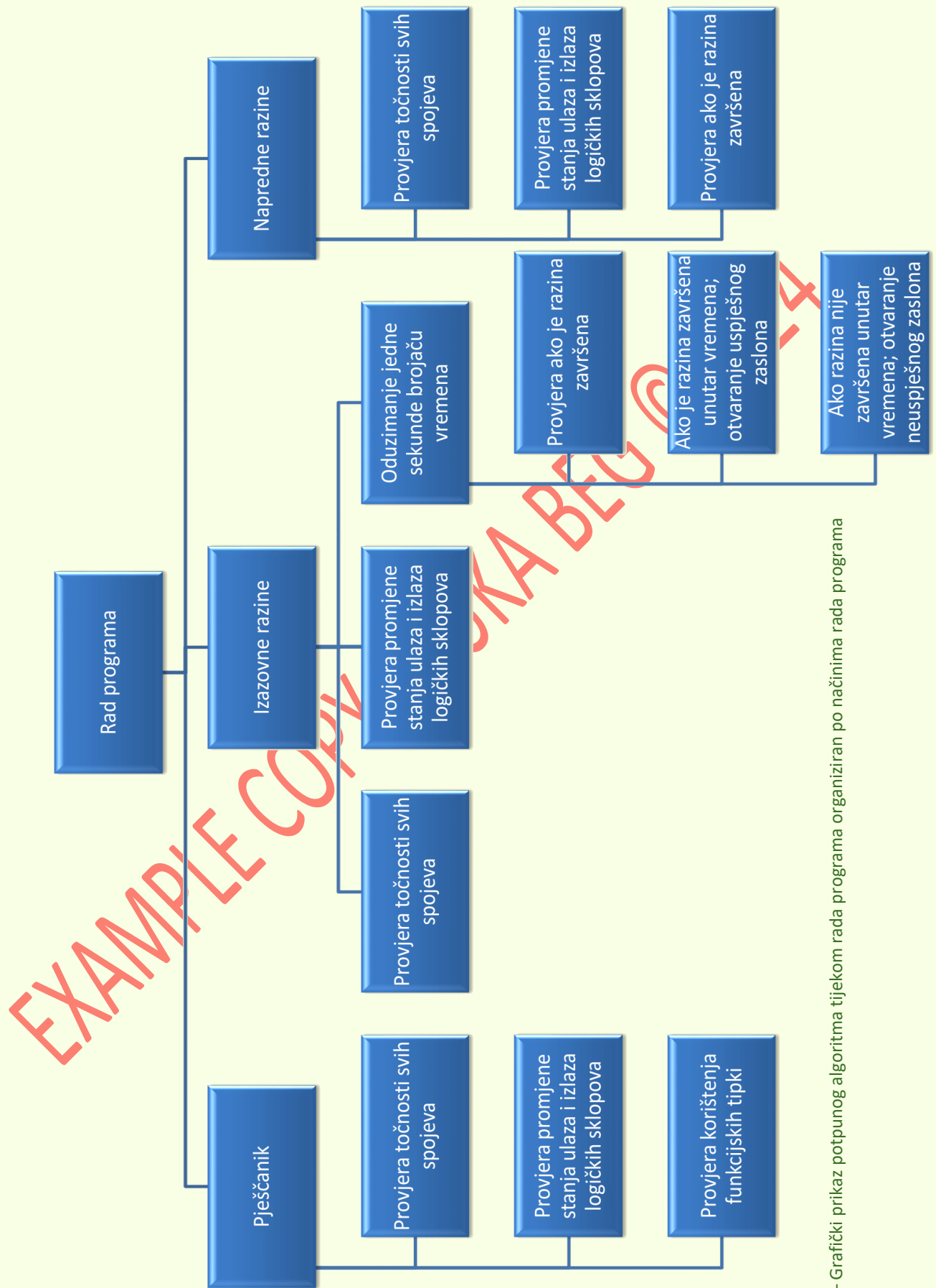
Logika sklopova



Slika 28. – Grafički prikaz algoritma logike logičkih sklopova u programu SLS



Slika 29. – Grafički prikaz potpunog algoritma programa pri otvaranju programa SLS



Slika 30. – Grafički prikaz potpunog algoritma tijekom rada programa organiziran po načinima rada programa

4.10. Popis C# skripti u programu

Naziv skripte	Deskripcija/Uloga	Kategorija
AndGate.cs	Osnovna logika za sklop I	Sklopovi
AndGateTruthTable.cs	Osnovna logika za sklop I u tablici istinitosti	Sklopovi Tablica istinitosti
AndGateTutorial.cs	Osnovna logika za sklop I u uputama	Sklopovi Upute
AudioManager.cs	Osnovni rad audio reproduktora Sadrži sve zvukove	Upravitelj Zvukovi
BuildData.cs	Razvojni podatci programa	Korisničko sučelje
BuildNumberTracker.cs	Stalna vrijednost inačice u programu	Upravitelj
ChallengeGateSpawner.cs	Stvaranje određenih sklopova u izazovnom načinu rada programa	Upravitelj Razine
ChallengeLevelData.cs	Spremanje podataka o razini	Razine
ChallengeLevelLoader.cs	Učitavanje podataka o razini	Razine
CheckIfInputFieldValid.cs	Provjera točnosti polja za unos	Korisničko sučelje
CloseWithESC.cs	Mogućnost zatvaranja prozora pritiskom na tipku Escape	Korisničko sučelje Upravitelj
CursorManager.cs	Promjena ikone pokazivača	Upravitelj Korisničko sučelje
DefaultSettings.cs	Sadrži sve zadane postavke rada programa	Upravitelj Postavke
DragAndDropSystem.cs	Osnovni rad Drag & Drop sistema u programu Koristi se za sklopove	Korisničko sučelje Upravitelj
ExitDialogLanguage.cs	Promjena jezika prozora za izlazak iz programa	Korisničko sučelje Jezici
FinishedLevelPanel.cs	Promjena jezika završnog ekrana u razinama Otvaranje uspješnog ili neuspješnog zaslona	Korisničko sučelje Upravitelj Razine Jezici
GameManager.cs	Sadrži sve važne funkcije, varijable i alate za opći rad programa	Upravitelj

GateInteraction.cs	Rad klika na gumb za ulaz ili izlaz sklopa	Sklopovi
GateStats.cs	Otvaranje razvojnih podataka o izabranom logičkom sklopu	Korisničko sučelje Sklopovi Razvoj
InputFieldArrowsFunction.cs	Rad „<“ i „>“ u polju za unos vrijednosti	Korisničko sučelje
InputGate.cs	Osnovna logika za ulaz logičkih sklopova	Sklopovi
InputGateTutorial.cs	Osnovna logika za ulaz u uputama	Sklopovi Upute
InputTruthTable.cs	Osnovna logika za izlaz u tablici istinitosti	Sklopovi Tablica istinitosti
IsMouseInside.cs	Provjera lokacije pokazivača Provjera ako je pokazivač unutar željenog objekta	Korisničko sučelje Upravitelj
LanguageClass.cs	Sadrži sva polja za tekstove tijekom promjene jezika	Jezici Upravitelj
LanguageEntries.cs	Sadrži učitane postavke jezika tijekom rada programa	Jezici Upravitelj
LanguageFileMaker.cs	Stvaranje datoteku sa spremljenim jezicima u JSON datoteku pri pokretanju programa	Jezici Upravitelj
LanguageLoader.cs	Učitavanje i dešifriranje JSON datoteke jezika pri pokretanju programa	Jezici Upravitelj
LanguageManager.cs	Sadrži vrijednosti izabranog jezika i standarda logičkih sklopova	Upravitelj Jezici Korisničko sučelje
LanguageTools.cs	Osnovne funkcije promjene jezika	Jezici Upravitelj
LevelCreator.cs	Čarobnjak za stvaranje razina	Razine Korisničko sučelje
LevelCreatorLanguage.cs	Promjena jezika čarobnjaka za stvaranje i učitavanje razina	Korisničko sučelje Razine Jezici
LevelInfo.cs	Otvaranje podataka izabrane razine	Razine Korisničko sučelje
LevelLoader.cs	Čarobnjak za učitavanje razina	Razine Upravitelj
LineDrawerManager.cs	Stvaranje žica te njihovo mijenjanje boja	Korisničko sučelje Upravitelj

LoadLevelObject.cs	Svaka razina u čarobnjaku za učitavanje razina	Korisničko sučelje Razine
LocalGateManager.cs	Sadrži logiku za brisanje ili dupliciranje sklopova	Upravitelj Sklopovi
MainMenu.cs	Osnovni rad početnog izbornika	Korisničko sučelje
MainMenuLanguage.cs	Promjena jezika početnog korisničkog sučelja	Korisničko sučelje Jezici
NamespaceSystem.cs	Sadrži sve važne varijable, kao npr. stanja i enum varijable za logičke sklopove	Upravitelj
NotGate.cs	Osnovna logika za sklop NE	Sklopovi
NotGateTruthTable.cs	Osnovna logika za sklop NE u tablici istinitosti	Sklopovi Tablica istinitosti
NotGateTutorial.cs	Osnovna logika za sklop NE u uputama	Sklopovi Upute
OrGate.cs	Osnovna logika za sklop ILI	Sklopovi
OrGateTruthTable.cs	Osnovna logika za sklop ILI u tablici istinitosti	Sklopovi Tablica istinitosti
OrGateTutorial.cs	Osnovna logika za sklop ILI u uputama	Sklopovi Upute
OutputGate.cs	Osnovna logika za izlaz logičkih sklopova	Sklopovi
OutputGateTutorial.cs	Osnovna logika za izlaz u uputama	Sklopovi Upute
OutputTruthTable.cs	Osnovna logika za ulaz u tablici istinitosti	Sklopovi Tablica istinitosti
PauseMenu.cs	Osnovni rad pauziranog izbornika	Korisničko sučelje
PauseMenuLanguage.cs	Promjena jezika pauziranog korisničkog sučelja	Korisničko sučelje Jezici
PuzzleLevelTake.cs	Uzima vrijednost integer varijable iz <i>PuzzleLevelIndexVar.cs</i>	Upravitelj Razine
PuzzleLoaderVar.cs	Sadrži vrijednosti izazovnih razina	Razine
RateWindow.cs	Promjena jezika prozora za ocjenjivanje rada programa	Korisničko sučelje Jezici
RightClickMenu.cs	Osnovne funkcije izbornika za desni klik	Korisničko sučelje Sklopovi

SafeArea.cs	Automatsko postavljanje objekata korisničkog sučelja unutar sigurne zone za telefone	Korisničko sučelje Upravitelj Pristupačnost
ScalableUIElement.cs	Mogućnost da neki dio korisničkog sučelja bude povećan/umanjen postavkom	Korisničko sučelje Pristupačnost
ScreenCaptureManager.cs	Slikanje trenutnog zaslona	Korisničko sučelje
ScreenshotDialog.cs	Otvaranje i zatvaranje skočnog prozora slikanja zaslona	Korisničko sučelje Upravitelj
SettingsData.cs	Sadrži sve moguće postavke u izborniku s postavkama rada programa	Postavke
SettingsMenu.cs	Osnovni rad izbornika s postavkama.	Korisničko sučelje Postavke
SettingsMenuLanguage.cs	Promjena jezika izbornika s postavkama rada programa	Korisničko sučelje Jezici Postavke
SettingsSaver.cs	Spremanje i učitavanje postavka u obliku datoteke	Postavke
SliderCounter.cs	Postavljanje vrijednosti klizača u tekst	Korisničko sučelje Postavke
SpawnerScript.cs	Rad gornje alatne trake - stvaranje logičkih sklopova	Korisničko sučelje Sklopovi Jezici
TruthTableCalculator.cs	Računanje vrijednosti tablice istinitosti	Tablica istinitosti Korisničko sučelje
TruthTableCell.cs	Unos vrijednosti izlaza i ulaza u tablicu istinitosti	Tablica istinitosti Korisničko sučelje
TruthTableLanguage.cs	Promjena jezika sučelja tablice istinitosti	Tablica istinitosti Jezici
TutorialMenuLanguage.cs	Promjena jezika u izborniku s interaktivnim uputama	Korisničko sučelje Jezici Upute
TutorialSelector.cs	Izbornik sklopova na dnu prozora interaktivnih uputa	Upravitelj Korisničko sučelje Upute

4.11. Jezici korisničkog sučelja u programu

SLS upravo sada podržava samo dva osnovna jezika: **Hrvatski jezik** i **Engleski jezik**. Svi prijevodi, tj. tekstovi, nalaze se u string varijablama koje se nalaze u skripti **LanguageClass.cs**

Zatim, one se pri svakom pokretanju programa stvaraju i pretvaraju u **JSON** datoteku koristeći funkcije programa **LanguageFileMaker.cs**, te učitavaju u memoriju programa koristeći program **LanguageLoader.cs**

Nakon stvaranja i učitavanja prijevoda, one se učitavaju u memoriju, tj. u statičnu skriptu **LanguageEntries.cs**, kako bi se mogle učitati i koristiti bez ponovnog učitavanja svih prijevoda iz originalne JSON datoteke.

Cilj mi je da u bliskoj budućnosti dodam sustav koji će omogućiti korisnicima da stvaraju svoje vlastite jezike i koriste ih u programu, te da na taj način jezici korisničkog sučelja nisu samo ograničeni na Hrvatski i Engleski jezik, već da su mogućnosti neograničene korisnicima diljem svijeta.

Prije nego sam dodao ovaj sustav, svi su se prijevodi već postavljali u skripte koje su bile namijenjene svakom dijelu korisničkog sučelja, te su se pri svakoj promjeni morale ponovo učitavati i postavljati.

4.12. Računanje tablice istinitosti

Pri svakoj ostvarenoj vezi između dva logička sklopa, ta dva logička sklopa spajaju se na dva različita mjesta. Prvo mjesto jest program tog logičkog sklopa koji sadrži osnovnu logiku za taj logički sklop, dok je drugo mjesto program tog logičkog sklopa koji je stvoren samo za svrhe računanja tablice istinitosti.

Prije računanja tablice istinitosti, prvo se provjerava broj mogućih kombinacija. Ako je broj kombinacija manji od 1, to znači da ne postoji ni jedan spoj između dva logička sklopa, pa je računanje tablice istinitosti nemoguće.

Ako je broj kombinacija veći od 1, stvaraju se sve moguće kombinacije, te u petlji se za svaki ulaz postavlja vrijednost (0 – neaktivni signal, 1 – aktivni signal), a zatim se očitava vrijednost izlaza. Između svakog postupka petlje ostavlja se 0.1 sekundi razlike kako ne bi slučajno došlo do neželjenog ometanja očitavanja od prošlih vrijednosti. Sve se očitane vrijednosti spremaju u listu.

Nakon računanja tablice istinitosti za svaku vrijednost stvara se element korisničkog sučelja u koji se unose vrijednosti izračunate tablice istinitosti.

5. Budućnost Simulatora Logičkih Sklopova

Razvoj ovog softvera nije završen! Prije svega, planiram proširiti mogućnosti logičkih sklopova u programu. Trenutačno su dostupni samo osnovni logički sklopovi poput **I**, **ILI**, **NE** te ulaza i izlaza, ali planiram dodati i naprednije sklopove kao što su **NI**, **NILI**, **XI**, **XILI** za one koji žele koristiti različite napredne logičke sklopove.

Također, planiram uvesti online ljestvicu bodova za različite izazovne i napredne razine. Uvest ću i korisničke račune kako bi korisnici mogli spremati svoje postavke i statistiku razina na svoj on-line račun. Planiram dodati i mogućnost multi-player¹³ načina rada programa tako da korisnici mogu s prijateljima i drugim ljudima dijeliti svoje ideje ili rješavati izazovne i napredne razine.

Kako bih potaknuo korištenje pješčanika u programu, dodat ću opciju spremanja i učitavanja razina pješčanika u datoteku poput izazovnih razina. Time će korisnici moći dijeliti svoje ideje s drugima i isprobavati ih na najlakši način.

Planiram komunicirati i s raznim udrugama i školama na području Karlovca i Karlovačke Županije, poput Udruge informatičara Karlovačke županije i Gradske knjižnice "Ivan Goran Kovačić", koje provode besplatne radionice informatike za mlade. Cilj mi je potaknuti korištenje svog programa u edukaciji i obrazovanju mlađih.

Također, želim testirati svoj softver s ljudima različitih dobnih skupina kako bih dobio povratne informacije o lakoći korištenja korisničkog sučelja i logičkih sklopova te dobio nove ideje za daljnji razvoj programa.

Cilj mi je objaviti svoj program na raznim platformama poput **Google Play**, **Apple Store** i **itch.io** kako bi i ostali ljudi mogli koristiti moj softver za obrazovanje i zabavu.

¹³ **Multi-player** – Korištenje programa uz ostale korisnike, više igrača

6. Zaključak

Tijekom samog razvoja softvera puno sam naučio o programiranju i stvaranju vlastitog softvera. Potrebno je uložiti velike količine truda i znanja. Puno sam novih stvari naučio o programiranju, poput korištenja raznih vrsti varijabli (enum i namespace), static vrijednosti i funkcija, Unity-evog **LineRenderera** te korištenja **Github** platforme za suradnju i praćenje razvoja programa.

Nadam se da će ovaj program pomoći mnogima pri stjecanju novih znanja o logičkim sklopovima te da će se u međuvremenu zabaviti tijekom korištenja.

Zahvaljujem se svojim roditeljima, svim prijateljima, te mentorima za poticaj i potporu pri izradi ovog softvera.

7. Literatura

1. **YouTube.** (2022, siječanj 15). Spremanje izazovnih/naprednih razina te postavki u datoteke na uređaj [Videozapis]. YouTube. Preuzeto sa https://www.youtube.com/watch?v=XOjd_qU2Ido&t=952s
2. **YouTube.** (2022, siječanj 22). Stvaranje čarobnjaka za instalaciju te exportanje projekta u .exe format [Videozapis]. YouTube. Preuzeto sa <https://www.youtube.com/watch?v=7nxKAtxGSn8>
3. **YouTube.** (2022, siječanj 30). Stvaranje crta/žica između logičkih sklopova (nije u potpunosti iskorišteno, puno je promijenjeno u programu) [Videozapis]. YouTube. Preuzeto sa <https://www.youtube.com/watch?v=5ZBynjAsfwI&t=65s>
4. **Unity Technologies.** (2020). Spremanje datoteka potrebnih za rad igre u .JSON datoteku (nadovezuje se sa prvom točkom). U Unity: Unity - Scripting API. Preuzeto sa <https://docs.unity3d.com/2020.1/Documentation/Manual/JSONSerialization.html>
5. **Unity Forum.** (2018, lipanj 15). Solution: LayoutElement fit parent with aspect ratio [Forum post]. Unity Forum. Preuzeto sa <https://forum.unity.com/threads/solution-layoutelement-fit-parent-with-aspect-ratio.542212/>
6. **Unity Technologies.** (2020). Promjena ikone pokazivača miša. U Unity: Unity - Scripting API. Preuzeto sa <https://docs.unity3d.com/ScriptReference/Cursor.SetCursor.html>
7. **Unity Technologies.** (2020). TextMeshPro (TMP) za tekstove i elemente korisničkog sučelja. U Unity: Unity - Scripting API. Preuzeto sa <https://docs.unity3d.com/Manual/com.unity.textmeshpro.html>

8. Sadržaj slika u tehničkoj dokumentaciji

Slika 1. – Luka Beg, Državno natjecanje iz Informatike 2023. godine	5
Slika 2. – Luka Beg, Državno natjecanje iz Informatike 2024. godine	5
Slika 3. – Luka Beg, Državno natjecanje iz Informatike 2024. godine, prezentacija rada SLS.....	6
Slika 4. – Logotip programa SLS	6
Slika 5. – Simbol za logički sklop NE (IEC/ANSI, NOT gate)	7
Slika 6. – Simbol za logički sklop ILI (IEC/ANSI, OR gate)	7
Slika 7. – Simbol za logički sklop i (IEC/ANSI, AND gate)	7
Slika 8. – Trello, web-stranica koju sam koristio tijekom razvoja softvera za razmjenjivanje ideja s mentorima te praćenja napretka	8
Slika 9. – Gumb u programu SLS za ulazak u pješčanik	9
Slika 10. – Gumb u programu SLS za ulazak u izazovni način rada programa	9
Slika 11. – Gumb u programu SLS za ulazak u napredni način rada programa	9
Slika 12. – Početni izbornik programa SLS.....	10
Slika 13. – Izbornik za odabir načina rada programa SLS	11
Slika 14. – Čarobnjak za stvaranje razina programa SLS.....	12
Slika 15. – Izbornik za učitavanje spremjenih razina programa SLS	13
Slika 16. – Izbornik s interaktivnim uputama za logičke sklopove.....	14
Slika 17. – Izbornik s postavkama rada programa SLS.....	15
Slika 18. – Razlika između IEEE/ANSI (lijevo) i IEC (desno) simbola logičkih sklopova	17
Slika 19. – Pješčanik programa SLS.....	19
Slika 20. – Izazovni način rada programa SLS.....	19
Slika 21. – Napredni način rada programa SLS.....	20
Slika 22. – Tablica istinitosti u pješčaniku	21
Slika 23. – Logički sklop ILI u programu SLS.....	21
Slika 24. – Lančani prikaz spajanja logičkih sklopova u programu	29
Slika 25. – Hijerarhija Unity tagova za razlikovanje vrsta logičkih sklopova u programu SLS	32
Slika 26. – Hijerarhija Unity tagova za razlikovanje ulaza (A i B) te izlaza logičkih sklopova u programu SLS	32
Slika 27. – Grafički prikaz algoritma za povezivanje logičkih sklopova u programu SLS	33
Slika 28. – Grafički prikaz algoritma logike logičkih sklopova u programu SLS	33
Slika 29. – Grafički prikaz potpunog algoritma programa pri otvaranju programa SLS.....	34
Slika 30. – Grafički prikaz potpunog algoritma tijekom rada programa organiziran po načinima rada programa.....	35