

# Sustav za prepoznavanje znakovne abecede

---

Jandrić, Ivan

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:190:501478>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2024-12-25**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Engineering](#)



SVEUČILIŠTE U RIJECI  
TEHNIČKI FAKULTET  
Diplomski sveučilišni studij računarstva

Diplomski rad

# Sustav za prepoznavanje znakovne abecede

Rijeka, srpanj 2024.

Ivan Jandrić  
0069078638

SVEUČILIŠTE U RIJECI  
**TEHNIČKI FAKULTET**  
Diplomski sveučilišni studij računarstva

Diplomski rad

# Sustav za prepoznavanje znakovne abecede

Mentor: prof. dr. sc. Miroslav Vrankić

Rijeka, srpanj 2024.

Ivan Jandrić  
0069078638

Rijeka, 22. ožujka 2023.

Zavod: **Zavod za automatiku i elektroniku**  
Predmet: **Asistivna tehnologija**  
Grana: **2.09.03 obradba informacija**

## ZADATAK ZA DIPLOMSKI RAD

Pristupnik: **Ivan Jandrić (0069078638)**  
Studij: Sveučilišni diplomski studij računarstva  
Modul: Programsko inženjerstvo

Zadatak: **Sustav za prepoznavanje znakovne abecede / Sign alphabet recognition system**

### Opis zadatka:

Predložiti realizaciju sustava za učenje znakovne abecede. Sustav bi sadržavao slike i opise za svako pojedino slovo znakovne abecede te mogućnost učenja pojedinačnog slova te slovanja riječi. Opisati postojeća rješenja takvih sustava. Objasniti arhitekturu predloženog sustava te opisati njegove prednosti i nedostatke.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

Zadatak uručen pristupniku: 22. ožujka 2023.

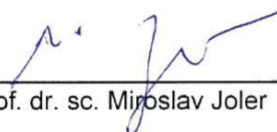
Mentor:



---

Prof. dr. sc. Miroslav Vrankić

Predsjednik povjerenstva za  
diplomski ispit:



---

Prof. dr. sc. Miroslav Joler

## Izjava o samostalnoj izradi rada

Izjavljujem da sam samostalno izradio ovaj rad.

Rijeka, srpanj 2024.

-----  
Ivan Jandrić

# Zahvala

Zahvaljujem mentoru prof. dr. sc. Miroslavu Vrankiću na mentorstvu i podršci tijekom pisanja ovoga rada. Zahvaljujem sestri Zlatnoj, roditeljima Zlatku i Lidiji te prijateljima na podršci tijekom studiranja.

# Sadržaj

<b>1</b>	<b>Uvod</b>	<b>1</b>
1.1	Motivacija . . . . .	1
1.2	Ciljevi diplomskog rada . . . . .	2
1.3	Organizacija diplomskog rada . . . . .	2
<b>2</b>	<b>Sustavi za učenje znakovne abecede</b>	<b>3</b>
2.1	Prepoznavanje znakovne abecede . . . . .	4
2.1.1	Prepoznavanje korištenjem rukavice . . . . .	4
2.1.2	Prepoznavanje korištenjem kamere . . . . .	5
2.1.3	MediaPipe Hands . . . . .	6
2.2	Komercijalno dostupne aplikacije . . . . .	7
2.2.1	Lingvano . . . . .	7
2.2.2	ASL Bloom . . . . .	8
2.2.3	Pocket Sign . . . . .	9
<b>3</b>	<b>Predloženi sustav</b>	<b>12</b>
3.1	Lekcije i zadatci . . . . .	13
3.1.1	Pitanje s više ponuđenih odabira . . . . .	14
3.1.2	Detekcija traženog znaka korištenjem kamere i umjetne inteligencije . . . . .	15

## Sadržaj

3.1.3	Spajanje znakova s njihovim značenjem . . . . .	15
3.2	Korisnički profil i statistika . . . . .	16
<b>4</b>	<b>Metodologija</b>	<b>18</b>
4.1	Prepoznavanje znakovne abecede . . . . .	18
4.1.1	Skup podataka . . . . .	18
4.1.2	Odbačeni vlastoručni model . . . . .	19
4.1.3	MediaPipe Gesture Recognizer model . . . . .	21
4.1.4	Optimizacija hiperparametara učenja . . . . .	22
4.2	Korisničko sučelje . . . . .	24
4.2.1	Osnovne struktura podataka . . . . .	25
4.2.2	Praćenje stanja aplikacije . . . . .	26
4.2.3	Integracija umjetne inteligencije . . . . .	26
4.2.4	Napredak i statistika . . . . .	28
4.2.5	Generiranje lekcija . . . . .	29
4.2.6	Međuspremnik slika . . . . .	31
<b>5</b>	<b>Rezultati</b>	<b>32</b>
5.1	Izgled aplikacije . . . . .	32
5.2	Postavke . . . . .	33
5.3	Odabir lekcije . . . . .	35
5.4	Lekcija i pitanja . . . . .	36
5.5	Slovkanje . . . . .	39
5.6	Statistika . . . . .	41
<b>6</b>	<b>Zaključak</b>	<b>43</b>
	<b>Literatura</b>	<b>45</b>



*Sadržaj*

<b>Popis slika</b>	<b>47</b>
<b>Popis tablica</b>	<b>49</b>
<b>Popis isječaka koda</b>	<b>50</b>
<b>Popis kratica</b>	<b>51</b>
<b>Sažetak</b>	<b>52</b>
<b>A Poveznice</b>	<b>53</b>

# Poglavlje 1

## Uvod

### 1.1 Motivacija

Prema podacima Svjetske federacije gluhih (engl. World Federation of the Deaf (WFD)), trenutno je u svijetu više od 70 milijuna gluhih i nagluhih osoba kojima je za komunikaciju neophodno korištenje znakovnog jezika.

Znakovni jezici su u potpunosti prirodni jezici s jasno definiranim pravilima, strukturom i abecedom; shodno tome, prirodni jezici često mogu imati više znakovnih jezika. (e.g. američki znakovni jezik i britanski znakovni jezik su namijenjeni kao zamjena za engleski jezik).[1]

Učenje prirodnih jezika je složen i (ovisno o učeniku) dugotrajan proces koji je (u slučaju učenja znakovnih jezika) dodatno otežan manjim brojem lako dostupnih materijala, pomoćnih alata i sl. Shodno tome, mali broj osoba bez poteškoća sa sluhom poznaje neki od znakovnih jezika. Jednostavnija alternativa učenju cijelog znakovnog jezika je učenje samo njegove abecede. Slovkanje korištenjem potonje omogućava sporazumijevanje znakovima pritom koristeći gramatiku i pravila klasičnog prirodnog jezika.

Sustavi za učenje kroz igru, tzv. gamificirani sustavi (engl. Gamification) mogu pozitivno utjecati na želju za učenjem te na sami ishod učenja.[2] Popularna aplikacija za učenje prirodnih jezika Duolingo, dokazano može svoje korisnike dovesti do samostalnog stupnja razumijevanja (B2) prema Zajedničkom europskom referent-

nom okviru za jezike. [3] Slične aplikacije postoje za učenje znakovnog jezika, ali se uglavnom plaćaju i/ili imaju manji broj dostupnih funkcionalnosti.

## **1.2 Ciljevi diplomskog rada**

Postojeće sustave za učenje znakovne abecede moguće je unaprijediti gamifikacijom i primjenom umjetne inteligencije. Primjena umjetne inteligencije u svrhu prepoznavanja znakovnog govora, omogućila bi lakše sporazumijevanje, prevođenje te učenje znakovnih jezika. Trenutno dostupni modeli za prepoznavanje znakovne abecede i metode strojnog učenja za razvoj istih, polučuju zadovoljavajuću razinu odziva (engl. Recall) i točnosti detekcije (engl. Accuracy).[4] Shodno tome, kroz ovaj diplomski rad bit će predložen i realiziran sustav za učenje znakovne abecede s elementima gamifikacije i strojnog učenja. Predloženi sustav će nastojati ukloniti ili umanjiti ograničenja postojećih rješenja, pritom omogućavajući nadograđivanje i proširenje u budućnosti.

## **1.3 Organizacija diplomskog rada**

Diplomski rad podijeljen je u šest poglavlja. U uvodnom poglavlju iznesen je problem te motivacija za njegovo rješavanje. Drugo poglavlje sadrži pregled trenutno dostupnih sustava, kako komercijalnih tako i sustava razvijenih u sklopu znanstvenih istraživanja. U trećem poglavlju će, kroz analizu ograničenja i nedostataka postojećih rješenja, biti predstavljen sustav za učenje znakovne abecede. Četvrto poglavlje, opisat će dostupne modele umjetne inteligencije za prepoznavanje znakovne abecede te metode strojnog učenja za razvoj istih. Na kraju četvrtog poglavlja bit će opisane odabrane metode te razlog za odabir istih. U petom poglavlju prikazat će se primjer sustava koji zadovoljava definirane ciljeve te implementiranog prethodno obrazloženim metodama. U posljednjem, šestom poglavlju rada, iznijet će se pregled ostvarenih ciljeva te zaključak diplomskog rada. Također, bit će navedena moguća poboljšanja predstavljenog rješenja.

## Poglavlje 2

# Sustavi za učenje znakovne abecede

Učenje prirodnih jezika je složen i dugotrajan proces. Optimizacija potonjeg procesa se kroz povijest primarno istraživala na području humanističkih znanosti (jezikoslovlje) i društvenih znanosti (psihologija). Početkom 20. stoljeća, primarna metoda učenja jezika temeljila se na biheviorizmu (engl. behaviour - ponašanje). Učenje se postiže imitacijom, što iziskuje čitanje i ponavljanje slova, riječi i/ili rečenica. Problem biheviorizma je što je u teoriji pasivan proces bez elemenata aktivnog učenja. U praksi, aktivno učenje se na podsvjesnoj razini postiže prevođenjem na već poznati jezik i usporedbom jezika koji se uči s već usvojenim jezicima. Lingvistički pristup učenju jezika razvio se sredinom 20. stoljeća kao odgovor na nedostatke biheviorističkog pristupa. U lingvističkom pristupu učenje se ne postiže imitacijom (ponavljanjem i pamćenjem), već aktivnim prevođenjem iz poznatog jezika u jezik koji se uči. Kroz proces prevođenja, učenik uči i pamti gramatička pravila jezika koji se uči. Međutim, istraživanja su pokazala kako aktivno učenje gramatičkih pravila nema potpunu korelaciju sa sposobnošću korištenja jezika u društvenom okruženju. Tom problemu nastojalo se doskočiti krajem 20. stoljeća, pretvaranjem lingvističkog pristupa u sociolingvistički pristup učenju. On se također bazira na aktivnom prevođenju i formiranju rečenica, međutim kurikulum je strukturiran po društvenim situacijama umjesto po gramatičkim pravilima. Sociolingvistički pristup uvelike ovisi o kvaliteti učitelja te njegovoj sposobnosti prilagodbe kurikuluma učeniku. Izazov sociolingvističkog pristupa je u potrebi kreiranja velikog broja društvenih situacija, njihovih varijacija te gradacije težine istih. Također, odabrane društvene situacije

moraju socioekonomski odgovarati okruženju u kojem se učenik nalazi.[5]

Sve snažniji računalni sustavi te napredak na području umjetne inteligencije i podatkovnih znanosti, mogu ublažiti prethodno opisane probleme učenja prirodnih jezika. Osim kvalitetno izrađenog kurikuluma, ograničavajući faktor pristupačnosti učenja prirodnih jezika je pristup kvalitetnim učiteljima. Učitelj provjerava učenikovo znanje te ovisno o ishodu provjere, mijenja kurikulum kako bi on bolje odgovarao učenikovim potrebama. Prvi računalni sustavi za učenje prirodnih jezika učitelja su nastojali "zamijeniti" algoritmima baziranim na dotadašnjoj heurističkoj analizi. Posljednjih godina, boljim rješenjem su se pokazali statistički modeli bazirani na povratnim neuronskim mrežama (engl. Recurrent Neural Network (RNN)), specifično mreže dugotrajno kratkoročne memorije (engl. Long short-term memory (LSTM)).[6]

## **2.1 Prepoznavanje znakovne abecede**

Glavna prepreka prepoznavanja te stoga i validacije govora znakovnim jezikom leži upravo u načinu na koji se govor znakovnim jezikom odvija; unutar vremena u trodimenzionalnom (3D) prostoru. Također, dodatan izazov čini manjak kvalitetnih i opsežnih baza podataka nad kojim bi se vršilo strojno učenje.[7] Razlikujemo dva glavna pristupa detekciji značajki ruke; pristup korištenjem specijalizirane rukavice (engl. Glove Based Approach (GBA)) i pristup korištenjem kamere (engl. Visual Based Approach (VBA)). [8] U tablici 2.1 je prikazana usporedba glavnih razlika prepoznavanja korištenjem rukavice i kamere.

### **2.1.1 Prepoznavanje korištenjem rukavice**

Prvi pokušaji detekcije znakovnog jezika bili su potkraj 20. stoljeća. Radi neadekvatne snage tadašnjih računala, takvi pokušaji uključivali su izradu specijaliziranog sklopovlja, najčešće rukavice sa sensorima, kako bi se dobio položaj značajki ruke unutar prostora. Rukavica u sebi sadrži veliki broj senzora koji daju izuzetno precizne podatke o točkama unutar prostora. Prednost ovakvog pristupa je izuzetno mala algoritamska zahtjevnost pripreme podataka što rezultira izuzetno velikom brzinom detekcije i velikom otpornošću na smetnje (i.e. šum). Mana su cijena izrade i

## Poglavlje 2. Sustavi za učenje znakovne abecede

praktičnost nošenja rukavice.[8].

Tablica 2.1 Usporedba GBA i VBA pristupa

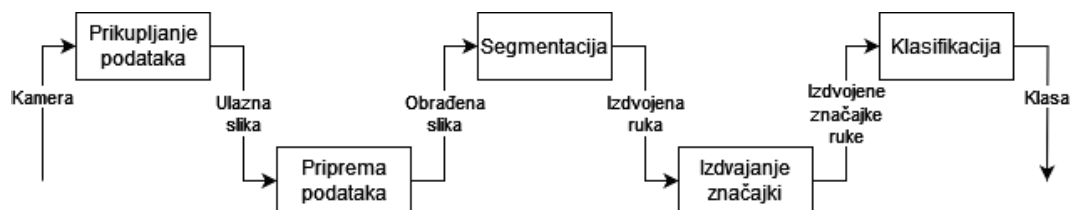
Faktor	GBA	VBA
Ulazni uređaj	Rukavica sa sensorima	Kamera
Brzina	Velika	Ovisi o implementaciji i jačini računala
Točnost	Ovisi o broju i kvaliteti senzora	Ovisi o rezoluciji i značajkama kamere
Smetnje u signalu	Male	Velike
Cijena	Visoka	Niska
Praktičnost	Mala	Velika

### 2.1.2 Prepoznavanje korištenjem kamere

Alternativni i u 21. stoljeću popularniji način prepoznavanja znakovnog govora iziskuje korištenje 2D ili 3D kamere. Mana i glavna prepreka korištenja kamere je izuzetno velika ovisnost o kvalitetno odrađenoj digitalnoj obradi ulaznog signala. Unatoč dobro odrađenoj obradi, kvaliteta detekcije uvelike ovisi o rezoluciji kamere, vrsti kamere i broju kamera te nije moguće eliminirati smetnje (i.e. šum) u signalu. Dodatne smetnje mogu prouzročiti razlike u osvjetljenju prostorije i boji kože učenika. Glavna prednost ovakvog pristupa je njegova cijena te praktičnost; pristup kamerama zadovoljavajuće rezolucije je značajno veći nego pristup specijaliziranim rukavicama.[8]

Koraci prepoznavanja znakovnog jezika opisani su na slici 2.1. Svaki korak ima više mogućih implementacija te trenutno ne postoji univerzalno rješenje bez očitih nedostataka. Trenutno najpopularnije metode za izdvajanje značajki i klasifikaciju bazirane su na korištenju konvolucijske neuronske mreže (engl. Convolution Neural Network (CNN)).[9]

## Poglavlje 2. Sustavi za učenje znakovne abecede

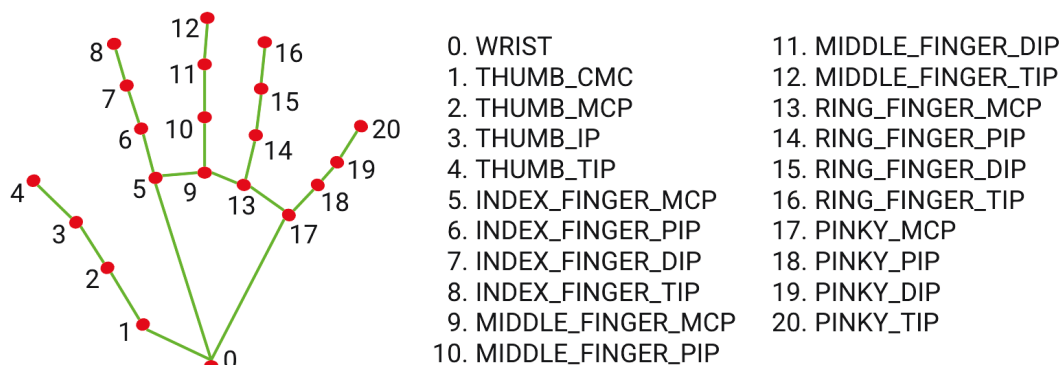


Slika 2.1 Koraci prepoznavanja znakovnog jezika

### 2.1.3 MediaPipe Hands

Istraživački laboratorij tvrtke Google razvio je knjižnicu pod imenom MediaPipe koja sadrži niz oglednih modela i alata za primjenu umjetne inteligencije. Knjižnica je primarno orijentirana na probleme vezane uz računalni vid. Za potrebe prepoznavanja znakovnog jezika osobito je zanimljiv model MediaPipe Hands čija je namjena prepoznavanje i praćenje značajki ruke.[10]

MediaPipe Hands je CNN model koji je u stanju prepoznati više ruku na slici, o kojoj ruci se radi (lijeva ili desna) te lokaliziranu i ne-lokaliziranu 21 značajku dlana. Značajke dlana prikazane su na slici 2.2.



Slika 2.2 Izdvojene značajke MediaPipe Hands modela

Izlaz modela moguće je koristiti za daljnje učenje te se u članku navodi primjer prepoznavanja gesti kao primjer moguće nadogradnje MediaPipe Hands modela.

## 2.2 Komercijalno dostupne aplikacije

Nažalost, niti jedan od istraženih, trenutno komercijalno dostupnih sustava za učenje znakovnih jezika nema mogućnost personalizacije učenja bazirane na strojnom učenju te nerijetko ne sadrže validaciju učenja potpomognutu umjetnom inteligencijom. Unatoč tome, u nastavku će biti ukratko opisani trenutno najpopularniji, komercijalno dostupni, sustavi za učenje znakovne abecede.

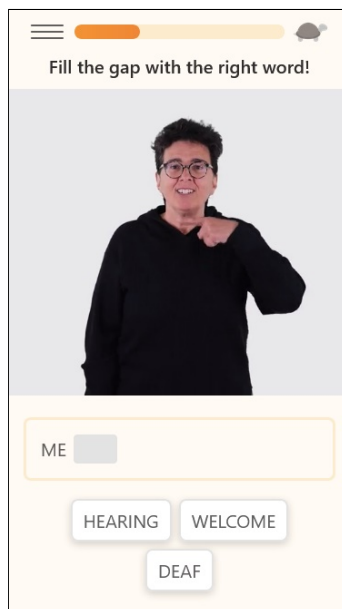
### 2.2.1 Lingvano

Lingvano je komercijalna aplikacija za učenje znakovnog jezika. Lingvano trenutno podržava učenje američkog znakovnog jezika (engl. American Sign Language (ASL)), britanskog znakovnog jezika (engl. British Sign Language (BSL)) te austrijskog znakovnog jezika (njem. Österreichische Gebärdensprache (ÖGS)). Učenje nije orijentirano samo na znakovnu abecedu, već na znakovni jezik u cijelosti. Učenje je zamišljeno kao skup poglavlja koja u sebi sadrže niz linearno poredani lekcija. Lekcije se sastoje od snimki učitelja koji pokazuju znak, riječ ili koncept koji se uči te od pitanja s višestrukim izborom odgovora. Primjer pitanja unutar lekcije prikazan je na slici 2.3. Lekcije je moguće ponavljati, ali je dostupna i opcija treninga (engl. trainer), gdje se ponavljaju pitanja na koja je korisnik prethodno krivo odgovorio. Osim načina na koji je učenje strukturirano, elementi gamifikacije također su uočljivi prisustvom praćenja dnevnog niza te nagrađivanjem točnosti. Aplikacija sadrži i "rječnik", gdje je na jednom mjestu moguće pregledati sve snimke učitelja za željeni znak.[11]

Lingvano je napravljen kao mobilna aplikacija pisana u JavaScript programskom jeziku (i/ili njegovim derivatima e.g. Typescripta) unutar Expo i React Native razvojnih okvira. Takav pristup omogućuje izuzetno laku izradu mobilne i mrežne varijante sučelja istovremeno. Nedostatak takvog pristupa je ograničenost na korištenje mrežnih tehnologija prilikom razvoja, što može rezultirati proizvodom nešto nižih performansi. Aplikacija ne sadrži detekciju znakova umjetnom inteligencijom.

Glavni nedostatak aplikacije je zahtijevanje registracije korisnika te nepostojanje besplatne verzije. U besplatnom (probnom) načinu, moguće je pristupiti samo prvih





Slika 2.3 Primjer pitanja unutar Lingvano aplikacije

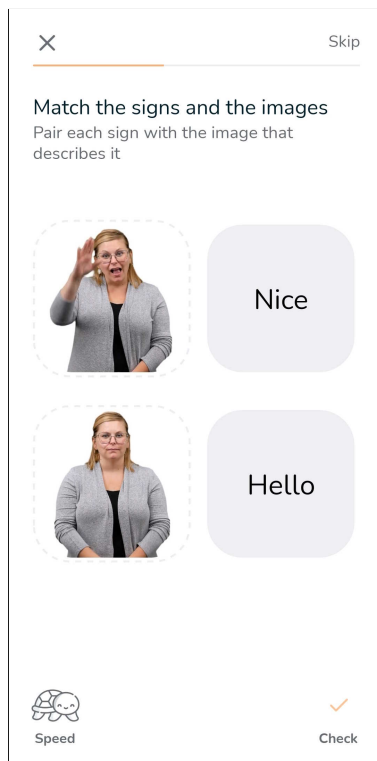
5 lekcija unutar prvog poglavlja. Također, pristup rječniku i treningu je ograničen unutar probnog perioda. Korisnik nema pristup detaljnoj statistici svog učenja.

### 2.2.2 ASL Bloom

ASL Bloom je komercijalna mobilna aplikacija za učenje znakovnog jezika. Princip rada aplikacije te struktura učenja su identični prethodno opisanoj aplikaciji Lingvano. Glavna razlika prilikom učenja uočljiva je u većoj raznovrsnosti pitanja. Osim običnih pitanja s višestrukim odabirom, prisutna su i točno/netočno pitanja te pitanja gdje je cilj spojiti snimku profesora s odgovarajućim značenjem. Primjer pitanja unutar lekcije prikazan je na slici 2.4. Gamifikacija je postignuta kroz skupljanje bodova rješavanjem. Količina sakupljenih bodova ovisi o točnosti rješavanja. Aplikacija također sadrži način vježbanja u vidu kviza ili flash kartica te "rječnik" gdje su dostupne snimke učitelja za željene znakove.[12]

Nažalost, tehnički opis aplikacije nije dostupan. Aplikacija je isključivo dostupna na mobilnim platformama iOS i Android, stoga je pretpostavka kako je rađena u

## Poglavlje 2. Sustavi za učenje znakovne abecede



Slika 2.4 Primjer pitanja unutar ASL Bloom aplikacije

višeplatformskoj (engl. multi-platform) tehnologiji kako bi se zadržala konzistentnost izvođenja.

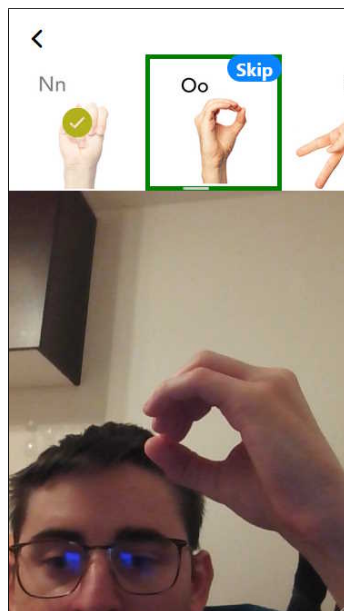
Glavni nedostatak aplikacije je, kao i kod Lingvano aplikacije, nepostojanje besplatne verzije. Tijekom probnog perioda dostupne su prve 3 cjeline te je dostupan pristup "rječniku" znakova. Sama aplikacija je dostupna isključivo na pametnim telefonima s iOS i Android operacijskim sustavima te nema dostupnu mrežnu verziju, čime nije dostupna na osobnim računalima. Korisnik nema pristup detaljnoj statistici svog učenja i Aplikacija je isključivo orijentirana na američki znakovni jezik.

### 2.2.3 Pocket Sign

Pocket Sign aplikacija dostupna je u obliku internetske stranice i mobilne aplikacije. Princip i struktura učenja unutar aplikacije također su slični prethodno opisanim

## Poglavlje 2. Sustavi za učenje znakovne abecede

aplikacijama. [13] Razlika u odnosu na prethodne aplikacije je dodatak opisnih pitanja unutar kojih je potrebno opisati pokazani niz znakova. Aplikacija također sadrži "rječnik" znakova. Gamifikacija je postignuta kroz prisutnost bodova. Količina prikupljenih bodova ovisi o točnosti rješavanja. Prikupljene bodove moguće je zamijeniti za "pojačanja" (engl. Power-ups) koja mogu pomoći prilikom rješavanja pitanja. Glavna prednost ove aplikacije je mogućnost ponavljanja uz pomoć umjetne inteligencije. Moguće je odabrati razne vježbe potpomognute umjetnom inteligencijom. Vježbe se sastoje od niza znakova koje je potrebno pokazati ispred kamere. Vježba je uspješno završena ako su svi znakovi ispravno pokazani. Primjer vježbanja s umjetnom inteligencijom prikazan je na slici 2.5.



Slika 2.5 Primjer umjetne inteligencije unutar Pocket Sign aplikacije

Koristeći razvojne alate preglednika moguće je izvršiti tehničku analizu aplikacije. Aplikacija je napravljena u JavaScript programskom jeziku (i/ili nekom od derivata JavaScript programskog jezika) koristeći React Native razvojni okvir. Prednosti i nedostaci takvog pristupa prethodno su opisani u sekciji 2.2.1. Detekcija dlana umjetnom inteligencijom izvedena je korištenjem MediaPipe Hands knjižnice [10]. Detaljan opis potonje nalazi se u poglavlju 4. Detektirane značajke dlana šalju

## *Poglavlje 2. Sustavi za učenje znakovne abecede*

se na poslužiteljski dio koristeći WebSocket protokol, uz kašnjenje od 200 milisekundi. Poslužiteljski dio napisan je u Python programskom jeziku te sadrži strojno naučeni modeli koji iz detektiranih značajki (kontura) dlana prepoznaje prikazani znak. Detaljnija analiza modela za prepoznavanje znaka nije moguća s obzirom na nemogućnost uvida u poslužiteljski kod. Pretpostavka autora ovog rada je kako se radi o modelu učenom na statičkim značajkama dlana.

Glavna mana aplikacije u odnosu na prethodno opisane aplikacije je zastarjeli izgled sučelja. Kao i u prethodno navedenim aplikacijama, mana su ograničenja unutar besplatne verzije aplikacije. Detekcija značajki dlana je zadovoljavajuća, ali modeli za prepoznavanje nisu u potpunosti ambidekstrični te je pojedine znakove moguće pokazati samo lijevom ili desnom rukom. Brzina detekcije uvelike ovisi o kvaliteti internetske mreže korisnika. Učenje korištenjem umjetne inteligencije odvojeno je od ostatka aplikacije te ne utječe na napredak unutar glavnog dijela aplikacije.

# Poglavlje 3

## Predloženi sustav

Na temelju istraživanja u prethodnom odjeljku, predloženi sustav trebao bi imati sljedeće glavne značajke:

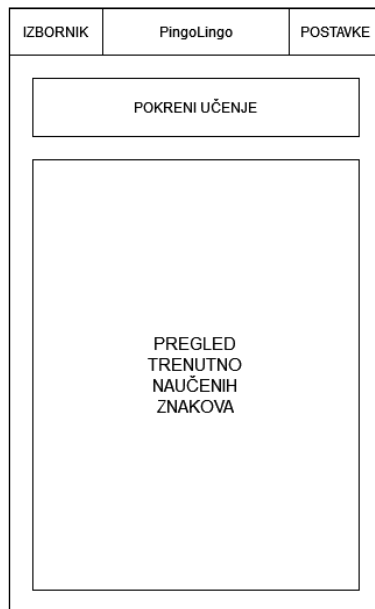
- Dostupnost neovisna o korisnikovoj platformi
- Dizajn prilagođen uređajima svih veličina
- Elementi gamifikacije (ciljevi, bodovi)
- Mogućnost promjene jezika i postavki učenja
- Korisnički profil s pamćenjem napretka
- Pregled detaljne statistike učenja

Kako bi sustav bio dostupan neovisno o platformi koju korisnik koristi, potrebno ga je razviti korištenjem višeplatformskih tehnologija. Za postizanje dizajna prilagođenog uređajima svih veličina, potrebno je dizajn orijentirati primarno na mobilne uređaje te ga zatim skalirati na uređaje većih veličina. Na vrhu sučelja uvijek bi se nalazila navigacijska traka (eng. NavBar - Navigation Bar) koja bi ovisno o trenutnom pogledu mijenjala svoj sadržaj. Radi pristupačnosti, korisnik bi u svakom trenutku korištenja aplikacije trebao moći pristupiti postavkama aplikacije.

Prema istraživanju učinkovitosti gamifikacije Duolingo platforme [14], na korisničko zadovoljstvo učenjem i učinkovitost učenja najviše utječu jasno definirani ciljevi napretka. Sučelje predloženog sustava sastojalo bi se od velikog gumba za pokretanje

### Poglavlje 3. Predloženi sustav

učenja te trenutnog napretka za svaki znak.



Slika 3.1 Skica glavnog sučelja

## 3.1 Lekcije i zadatci

Sustav treba omogućiti 3 vrste lekcija:

- Učenje slova
- Učenje brojeva
- Učenje riječi (slovkanje)

Na temelju istraživanja postojećih komercijalnih aplikacija u sekciji 2.2, poželjna značajka tijekom učenja je izmjenjivanje više vrsta zadataka kako bi se umanjila monotonost učenja. Sustav treba omogućiti rješavanje 3 vrste zadataka:

- Pitanje s više ponuđenih odabira
- Detekcija traženog znaka korištenjem kamere i umjetne inteligencije

- Spajanje znakova s njihovim značenjem

Kako učenje ne bi trajalo predugo, svaka lekcija trebala bi se sastojati od maksimalno 15 zadataka. Broj zadataka određen je empirijski po uzoru na postojeća rješenja. Osim izmjenjivanja različitih vrsta zadataka, količina svake vrste zadataka mora biti nasumično izabrana između donje i gornje granice, kako bi se uvela dodatna raznolikost tijekom učenja.

### 3.1.1 Pitanje s više ponuđenih odabira

Zadatak u obliku pitanja s više ponuđenih odabira, ali samo jednim točnim odabirom, prisutan je u svim aplikacijama istraženima u odjeljku 2.2. Predloženi dizajn pitanja s više ponuđenih odabira, prikazan je na slici 3.2. Dizajn je izuzetno fleksibilan jer se može i izokrenuti te bi u tom slučaju korisniku bilo prikazano slovo (ili broj) te bi se odabirao traženi znak. Ovakva vrsta zadatka može se primijeniti i za učenje slovkana. Umjesto običnih slika postoji mogućnost korištenja animacija u GIF (engl. Graphics Interchange Format (GIF)) formatu.



Slika 3.2 Skica sučelja pitanja s više ponuđenih odabira

### 3.1.2 Detekcija traženog znaka korištenjem kamere i umjetne inteligencije

Po uzoru na aplikaciju Pocket Sign [13], unutar sustava za učenje bili bi prisutni zadatci gdje je korisnikov cilj pokazati traženi znak ispred kamere. Nadogradnja u odnosu na prethodno spomenutu aplikaciju bilo bi jasnije sučelje s jasno definiranom trenutnom statistikom detekcije. Strojno učeni model treba biti upotrebljiv s obje ruke. Ovakva vrsta zadatka se može primijeniti i za učenje slovanja. Predloženi dizajn zadatka prikazan je na slici 3.3.



Slika 3.3 Skica sučelja pitanja s kamerom

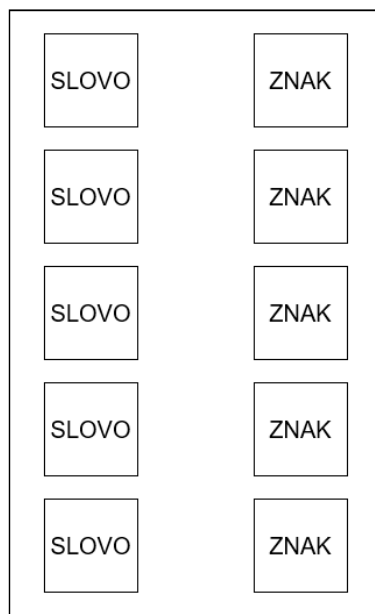
### 3.1.3 Spajanje znakova s njihovim značenjem

Zadatak spajanja znakova s njihovim značenjem prisutan je u aplikaciji ASL Bloom, međutim ograničen je na samo 2 spajanja (2 znaka s lijeve strane i 2 značenja s desne). [12] Predloženi sustav proširio bi broj spajanja na 5. S povećanim brojem spajanja povećao bi se i broj dopuštenih grešaka s 1 dopuštene greške na 3 dopuštene



### Poglavlje 3. Predloženi sustav

greške. Ovakva vrsta zadatka nije primjenjiva za učenje slovkana. Predloženi dizajn pitanja spajanja znakova s njihovim značenjem, prikazan je na slici 3.4.



Slika 3.4 Skica sučelja pitanja spajanja znakova s značenjem

## 3.2 Korisnički profil i statistika

Sustav treba sadržavati detaljan pregled korisnikove statistike. Sustav mora moći pamtit i korisnikov napredak. Gamifikacija napretka prisutna je u vidu ostvarenih bodova te trenutnom nizu (eng. streak) rješavanja.

Na temelju istraživanja u sekciji 2.2, uočen je nedostatak detaljne statistike korisničkog napretka. U istraživanju gamifikacije platforme Duolingo [14], mogućnost prikaza detaljne statistike trenutnog učenja može povećati korisnikov interes za učenjem jezika. Sukladno tome, sustav treba sadržavati mogućnost prikaza detaljne statistike učenja. Statistika mora biti prikazana na grafikonima te sadržavati podjelu na temelju vremena i vrste učenja.

Primjer sučelja korisničkog profila i statistika prikazan je na slici 3.5.

Poglavlje 3. Predloženi sustav



Slika 3.5 Skica sučelja korisničkog profila i statistike

# Poglavlje 4

## Metodologija

Kako bi se izradio sustav naveden u prethodnoj cjelini isprobano je više metoda prepoznavanja znakovne abecede te više tehnologija izrade korisničkog sučelja. U nastavku će biti navedene kako odabrane, tako i isprobane metodologije za prepoznavanje znakovne abecede i izradu korisničkog sučelja.

### 4.1 Prepoznavanje znakovne abecede

Za prepoznavanje kostura ruke i značajki dlana korištena je knjižnica MediaPipe Hands razvijena od strane istraživača tvrtke Google.[10] S obzirom na složenost prepoznavanja cijelog znakovnog jezika te kako je za potpuno prepoznavanje potreban holistički model, odnosno model cijelog tijela, izrađeni sustav usredotočen je samo na prepoznavanje brojeva i znakova unutar znakovne abecede.

#### 4.1.1 Skup podataka

Za učenje su korišteni već dostupni gotovi skupovi podataka preuzeti sa stranice Kaggle.

Za učenje slova američke znakovne abecede korišten je skup podataka s preko 87000 slika u 200 x 200 formatu. U skupu podataka klasa "nothing" preimenovana je u "none" kako bi bila konzistentna s oznakama koje koristi MediaPipe knjižnica.

## Poglavlje 4. Metodologija

Dodatne klase znaka razmaka i znaka za brisanje su ostavljene u skupu podataka, jer ne utječu na točnost prepoznavanja 26 znakova engleske abecede.[15] Prilikom testiranja, uočeni su određeni problemi u prepoznavanju određenih znakova te je učenje ponovljeno na skupu podataka s 27000 slika u 512 x 512 formatu. U skupu podataka klasa "blank" preimenovana je u "none" kako bi bila konzistentna s oznakama koje koristi MediaPipe knjižnica.[16] Oba modela dostupna su u završnoj aplikaciji.

Za učenje brojeva američke znakovne abecede korišten je skup podataka s preko 11000 slika u 512 x 512 formatu. U skupu podataka klasa "blank" preimenovana je u "none" kako bi bila konzistentna s oznakama koje koristi MediaPipe knjižnica.[17]

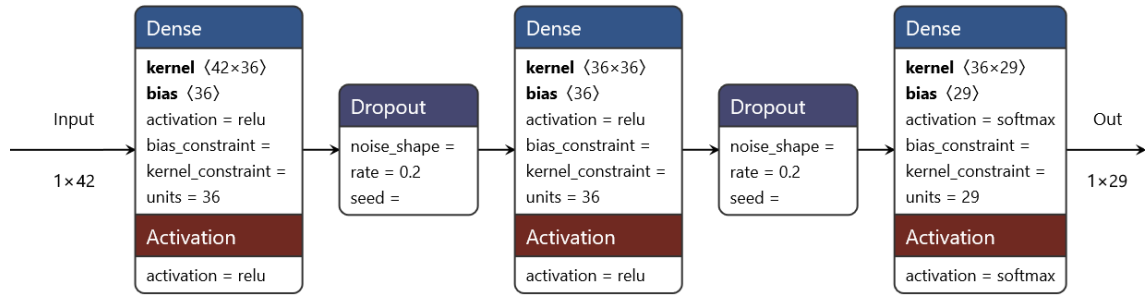
### 4.1.2 Odbačeni vlastoručni model

Za prepoznavanje značajki ruke koristi se MediaPipe Hands model koji je prethodno opisan u odjeljku 2.1.3. Za klasifikaciju tih značajki u konkretna imena pokazanih znakova, korišten je model baziran na modelu za prepoznavanje gesta za upravljanje uređajima.[18]

Struktura modela nalazi se na slici 4.1. Model se sastoji od 6 slojeva i 3953 parametra nad kojima je učeno. Ulazni sloj se sastoji od lokaliziranih X i Y koordinata 21 značajke dlana MediaPipe Hands modela, što sveukupno čini 42 značajke ulaznog sloja. Y koordinata nije korištena, jer MediaPipe Hands model nije u stanju u potpunosti prepoznati dubinu. Desnorukost/ljevorukost je ignorirana. Model sadrži 2 skrivena sloja koji se sastoje od međusobno povezanih neurona s ReLU aktivacijskom funkcijom i regularizacijom ispadanjem. Prvi skriveni sloj sadrži identičan broj neurona, dok drugi sadrži broj neurona jednak geometrijskoj sredini veličine ulaza i izlaza.

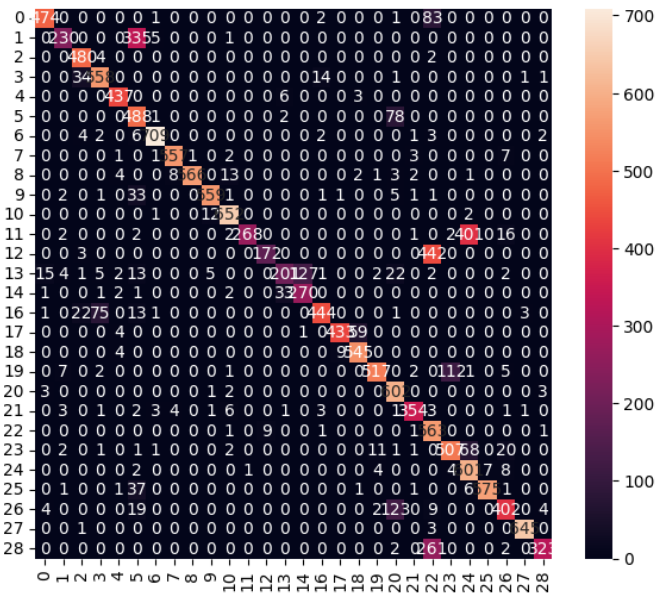
Model je tijekom validacije postigao preciznost od 87.7 %, što je relativno zadovoljavajući rezultat. Međutim, daljnjom analizom te testiranjem modela uočen je nedostatan odziv (engl. recall) modela, odnosno, model je prestrog te tijekom normalne uporabe ne može prepoznati sve znakove. Iz matrice zbunjenosti prikazane na slici 4.2, uočljivo je kako često dolazi do krivih klasifikacija pretežito sličnih znakova. Primjerice slovo B (index 1) se često krivo klasificira kao slovo F (index 5) jer se oba znaka sastoje pretežito od otvorenog dlana.

Poglavlje 4. Metodologija



Slika 4.1 Struktura odbačenog modela

Širenje skrivenih slojeva te povećanje dubine samog modela nije pomoglo u rješavanju uočenih nedostataka. Odluka o nekorištenju informacije o ljevorukosti i Z koordinati značajki u ulaznom sloju utječe na sposobnost razlikovanja sličnih znakova. Daljnje istraživanje i promjene na spomenutom modelu nisu vršene, već je trud preusmjeren na rješenja i alate koje nudi ostatak MediaPipe knjižnice.



Slika 4.2 Matrica zbunjenosti odbačenog modela

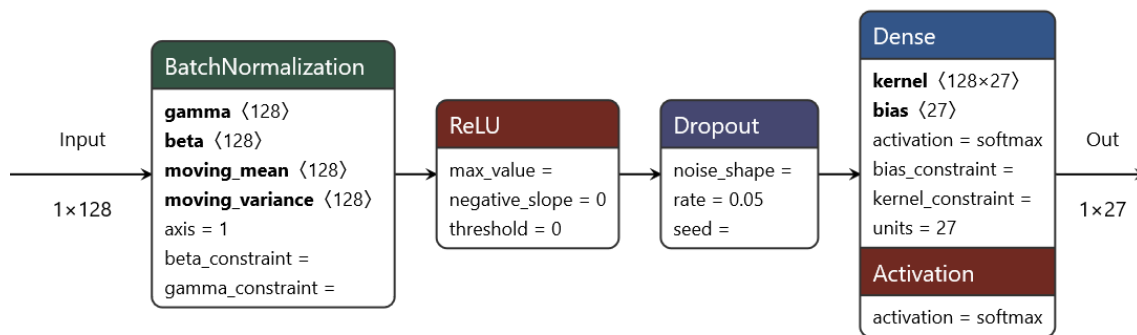
### 4.1.3 MediaPipe Gesture Recognizer model

Prethodno opisani model nije zadovoljavao preciznošću i odzivom. Za potrebe prepoznavanja gesti, MediaPipe tim razvio je ogledni model te pripadajući alat za prilagođavanje modela MediaPipe Model Maker. Sam alat pisan je u Python programskom jeziku te ga je moguće koristiti unutar Jupyter bilježnice. Kako bi MediaPipe Model maker mogao ispravno učitati skup podataka, podatke za učenje potrebno je organizirati unutar mapa, gdje svaka mapa predstavlja ime oznake koja će se naučiti. Za potrebe učenja, validacije i testiranja, učitan skup podataka dijeli se na sljedeći način:

- 80% podaci za učenje
- 10% podaci za validaciju
- 10% podaci za testiranje

Tijekom učenja moguća je promjena hiperparametara modela (poput broja i širine skrivenih slojeva neuronske mreže) te hiperparametara učenja. Hiperparametri modela nisu mijenjani dok je optimizacija hiperparametara učenja obrazložena u odjeljku 4.1.4.

Model za prepoznavanje znakovne abecede sastoji se od 5 slojeva čija je struktura prikazana na slici 4.3. Sam model sadrži 3995 parametara, od kojih se 3739 koristi za učenje.



Slika 4.3 Struktura prilagođenog Gesture Recognizer modela

## Poglavlje 4. Metodologija

Ulazni sloj sastoji se od 128 parametara, odnosno značajki dlana koje vraća MediaPipe Hands model. Redom to su:

- (3 x 21) Lokalizirane (X, Y, Z) koordinate 21 značajke dlana
- (3 x 21) Ne-lokalizirane (X, Y, Z) koordinate 21 značajke dlana
- (1 x 2) Desnorukost/ljevorukost te vjerojatnost ispravne detekcije potonje

Ulaz se normalizira nad grupom te šalje u skriveni sloj koji se sastoji od neurona s ReLU aktivacijskom funkcijom. Skriveni sloj je regulariziran ispadanjem. Izlazni sloj se sastoji od 27 neurona (26 slova abecede + 1 prazna detekcija) koji koriste softmax aktivacijsku funkciju.

MediaPipe Model Maker omogućava dodavanje skrivenih slojeva te povećavanje širine slojeva, međutim, s obzirom na zadovoljavajuću preciznost dobivenih rezultata to nije bilo potrebno.

### 4.1.4 Optimizacija hiperparametara učenja

MediaPipe Model Maker nudi mogućnost promjene hiperparametara učenja. Određivanje idealnih hiperparametara određeno je empirijski kroz usporedbu preciznosti i vrijednosti funkcije gubitka (engl. loss function). Broj epoha (iteracija) nije mijenjan, već je pušten na preporučenih 10, jer polučuje zadovoljavajuće rezultate bez pretjerane prilagodbe podacima. Razmatrani su sljedeći parametri:

- **learningRate** - Stopa učenja za gradijentni spust; početna vrijednost 0.001
  - Prevelika stopa učenja može dovesti do pretjerane prilagodbe podataka, dok premala može dovesti do premale prilagodbe podacima
  - U ekstremnim slučajevima, prevelika ili premala stopa učenja može dovesti do nemogućnosti konvergencije gradijentnog spusta
  - Početna vrijednost je 0.001
- **batchSize** - Broj uzoraka tijekom propagacije (unaprijed i unatrag)
  - Veća vrijednost povećava brzinu učenja, ali smanjuje preciznost učenja

## Poglavlje 4. Metodologija

- Početna vrijednost su 2 uzorka
- **lrDecay** - Opadanje stope učenja nakon svake iteracije
  - Što je broj bliže jedinici opadanje je sporije, odnosno, što je bliže nuli opadanje stope učenja je brže
  - Početna vrijednost je 0.99
- **gamma** - Parametar fokusiranja dinamičke unakrsne entropije (engl. focal loss)
  - Što je vrijednost veća to se veći značaj (fokus) daje krivo prepoznatim primjerima u odnosu na točno prepoznate primjere
  - Prevelika vrijednost može smanjiti točnost detekcije, jer će se učenje fokusirati na prepoznavanje rubnih slučajeva
  - Vrijednost 0 znači kako nema "fokusiranja" te je funkcija gubitka obična unakrsna entropija
  - Početna vrijednost je 2

Rezultati isprobavanja parametara nalaze se u tablici 4.1. Unutar tablice, gdje nije izričito napisana izmijenjena vrijednost, pretpostavlja se korištenje početnih vrijednosti. Podebljana vrijednost u stupcu parametara označava kako se radi o početnoj vrijednosti parametra (istaknuto za svrhu uspoređivanja). Podebljana vrijednost u stupcu gubitak i preciznost označava "najbolju" vrijednost.

Sva učenja provedena su na skupu podataka koji sadržava samo slova američke znakovne abecede. Iz priloženih rezultata uočljivo je kako početne vrijednosti stope učenja i opadanja stope učenja ishode dobre rezultate učenja. Početni gamma parametar rezultira najvećom preciznosti učenja, međutim, funkcija gubitaka je manja kada se gamma parametar poveća. Iz tog razloga tijekom učenja završnog modela korišten je gamma parametar vrijednosti 5.

Rezultati promjene broja uzoraka tijekom propagacije su obrnuti od očekivanih rezultata. Povećanje broja uzoraka rezultiralo je manjom vrijednosti funkcije gubitka te izuzetno velikom preciznošću. S obzirom na nelogičnost rezultata, dobiveni modeli



Tablica 4.1 Rezultati promjene hiperparametara učenja

Početne postavke uz...	Gubitak	Preciznost
learningRate = 0.01	0.447	85.7%
<b>learningRate = 0.001</b>	<b>0.199</b>	<b>93.5%</b>
learningRate = 0.0001	0.219	93.2%
<b>batchSize = 2</b>	0.199	93.5%
batchSize = 4	0.112	95.4%
batchSize = 8	0.069	97%
batchSize = 16	<b>0.052</b>	<b>97.4%</b>
lrDecay = 0.5	0.641	61.6%
<b>lrDecay = 0.99</b>	<b>0.199</b>	<b>93.5%</b>
Gamma = 0	0.362	91.1%
<b>Gamma = 2</b>	0.199	<b>93.5%</b>
Gamma = 5	<b>0.131</b>	90.8%

su isprobani unutar sustava s kamerom te je ispostavljeno kako se radi o modelima koji su pretjerano prilagođeni podacima (engl. over-fit). Sukladno tome, broj uzoraka tijekom učenja završnog modela postavljen je na početnu vrijednost.

## 4.2 Korisničko sučelje

S obzirom na to kako zamišljeni sustav treba biti dostupan neovisno o platformi te treba podržavati uređaje svih veličina, korisničko sučelje izrađeno je kao jednostranična web aplikacija (engl. Single Page Application (SPA)). Za izradu jednostranične web aplikacije korišten je TypeScript programski jezik i React knjižnica za izradu pogleda (engl. views). Za stil i izgled aplikacije korištene su već gotove komponente iz knjižnice komponenti Mantine, koje su po potrebi stilizirane koristeći ručno napisana CSS pravila. Sama aplikacija podržava prijevod sučelja te je prevođenje izvedeno korištenjem knjižnice Talkr.

## 4.2.1 Osnovne struktura podataka

Znakovni jezici unutar aplikacije predstavljeni su kao tip podataka tečaj. Svaki tečaj sadrži jedinstveni identifikacijski broj, ime (u obliku znakovnog niza za prijevod), sliku, listu strojno naučenih modela te listu znakova. Primjer strukture je prikazan u kodnoj listi 4.1. Ovakvom strukturom omogućeno je dodavanje dodatnih znakovnih jezika u budućnosti.

Isječak koda 4.1 Struktura tečaja

---

```
export type TCourse = {  
  id: ECourseIds  
  name: string  
  imagePath: string  
  models: TCourseModel []  
  symbols: TCourseSymbol []  
}
```

---

Svaki znak unutar tečaja sadrži jedinstveni identifikacijski broj, ime (u obliku znakovnog niza za prijevod), sliku slova/broja koji označava te više slika ili animacija prikaza znaka. Primjer strukture je prikazan u kodnoj listi 4.2. Ovakva struktura je odabrana kako bi bilo moguće imati više primjera pokazivanja znaka.

Isječak koda 4.2 Struktura znaka

---

```
export type TCourseSymbol = {  
  id: string  
  name: string  
  imagePath: string  
  handImagePaths: string []  
  isNumber: boolean  
}
```

---

## 4.2.2 Praćenje stanja aplikacije

Upravljanje stanjem (engl. state management) aplikacije izvedeno je korištenjem Zustand knjižnice za upravljanje stanjem. Sastoji se od 4 zasebna "skladišta" (engl. storage) koja ovisno o namjeni sadrže različite podatke i imaju različito vrijeme trajanja (privremeno ili trajno). U Tablici 4.2 prikazane su značajke korištenih "skladišta".

Tablica 4.2 Skladišta i njihove namjene

Skladište	Namjena	Vrsta pohrane
Postavki	Spremanje postavki (jezik sučelja, tema sučelja i sl.)	Trajna
Statistike	Spremanje napretka, broja bodova i ostale statistike	Trajna
Lekcije	Praćenje koraka, broja života i drugih varijabli vezanih za trenutno aktivnu lekciju	Privremena
Pred memorije	Držanje slika u memorija kako bi se ubrzalo učitavanje istih	Privremena

S obzirom na vrstu pohrane, skladišta su trajna ili privremena. Privremena skladišta se brišu prilikom izlaska aplikacije, dok se trajna skladišta pohranjuju u lokalni spremnik web preglednika. Prednost ovakvog pristupa je što korisnik nije dužan kreirati korisnički račun za pohranu u oblaku, ali velika mana je što je napredak vezan za preglednik unutar kojeg se aplikacija koristi te trenutno nije moguć izvoz i učitavanje napretka.

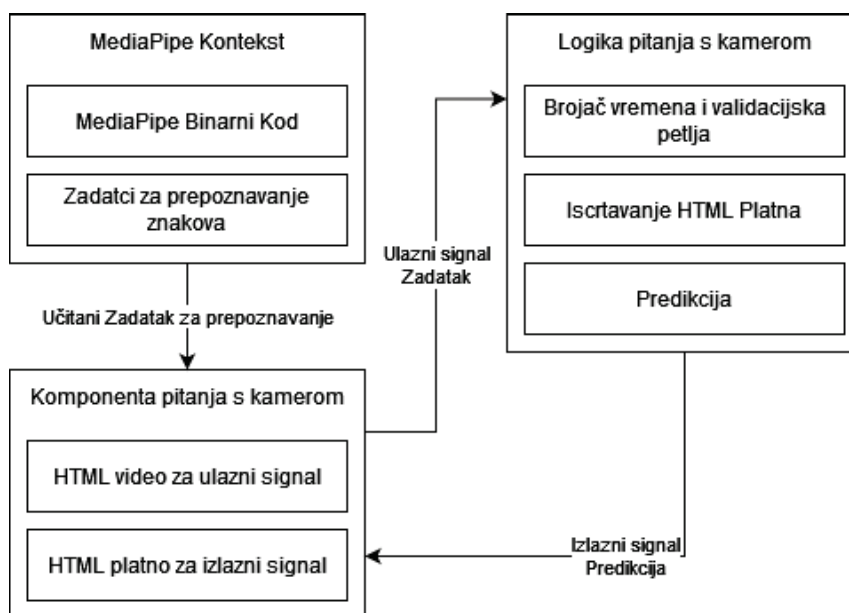
## 4.2.3 Integracija umjetne inteligencije

Za integraciju strojno naučenog modela iz odjeljka 4.1 u korisničko sučelje, korištena je JavaScript verzija MediaPipe knjižnice. Za ispravan rad knjižnice potrebno je dinamički učitati MediaPipe Tasks Vision binarni kod. Kako ne bi došlo do neočekivanih grešaka tijekom izvođenja, učitavanje binarnog koda se izvršava pri učitavanju aplikacije unutar React konteksta. Kada je binarni kod MediaPipe Tasks Vision

## Poglavlje 4. Metodologija

knjižnice učitani, potrebno je učitati strojno naučene zadatke za prepoznavanje znakovnog jezika.

Kompletno učitavanje i logika, nalazi se unutar React konteksta (engl. React Context), koji služi za umetanje zavisnosti (engl. dependency injection), odnosno objekata i funkcija u komponente koje zavise o njima. Pojednostavljeni prikaz integracije umjetne inteligencije i logike pitanja s kamerom prikazan je na slici 4.4. Na slici je uočljiva ovisnost komponente pitanja s kamerom o React kontekstu (na slici MediaPipe Kontekst), iz kojeg je preuzet binarni kod zadatka za prepoznavanje znakova.



Slika 4.4 Prikaz logike pitanja s kamerom

Pitanje s kamerom ne isctava izravno ulazni signal video kamere korisnika u HTML video komponentu, već je ulazna HTML video komponenta sakrivena, a njen ulazni signal se obrađuje te isctava na HTML platno. Ovakav pristup se koristi kako bi se omogućilo isctavanje kostura ruke i ključnih točaka na signalu video kamere. Ulazni signal i strojno naučeni zadatak za prepoznavanje znakovnog jezika šalju se u zaseban objekt koji sadrži petlju za validaciju i isctavanje HTML platna. Maksimalna brzina isctavanja i predviđanja u teoriji je ograničena na jednu milisekundu,

a u praksi ovisi o brzini osvježavanja korisnikova ekrana. Vremenska razlika između vremena prošle i trenutne predikcije se pamti u brojaču vremena te ju je potrebno proslijediti MediaPipe knjižnici za ispravno predviđanje. Predikcija se sastoji od predviđenog znaka, ruke na kojoj je znak pokazan te koeficijenta povjerenja.

#### 4.2.4 Napredak i statistika

Tijekom rješavanja svaki se pokušaj rješavanja pitanja, neovisno o uspješnosti, pamti i sprema u polje pokušaja. Trenutno je lokacija spremanja lokalni spremnik web preglednika, ali je implementacija fleksibilna te omogućava kasniju nadogradnju na spremanje u oblaku. Svaki pokušaj rješavanja u sebi sadržava kategoriju pitanja (slova, brojevi ili riječi), tip pitanja (višestruki izbor, pitanje spajanja, pitanje s kamerom itd.), jedinstvene identifikacijske brojeve znakova koji se pojavljuju u pitanju, vremensku oznaku (engl. timestamp) i uspješnost rješavanja. Primjer strukture je prikazan u kodnoj listi 4.3.

Isječak koda 4.3 Struktura pokušaja rješavanja

---

```
export type TQuestionAttempt = {  
  category: TQuestionCategory  
  symbolIds: string []  
  timestamp: number  
  type: TQuestionType  
  success: boolean  
}
```

---

Sva statistika i osvojeni bodovi, računa se iz navedenog polja. Znak se smatra uspješno naučenim kada je točno riješen barem 100 puta, odnosno kada se njegov jedinstveni identifikacijski broj nađe u 100 uspješnih pokušaja. Osvojeni broj bodova se pamti zasebno, ali je proporcionalan broju uspješnih pokušaja rješavanja. Pamti se zasebno, jer takav pristup omogućava veću fleksibilnost za primjenu bonus bodova, oduzimanje bodova i sl.

Trenutni broj aktivnih dana u nizu (engl. active days streak), računa se na sljedeći način:

## Poglavlje 4. Metodologija

1. Vremenska oznaka svakog pokušaja pretvara se u vremensku oznaku početka dana pokušaja
2. Svi pokušaji grupiraju se po vremenskoj oznaci početka dana te se iz dobivenog polja miču duplikati
3. Ako veličina novog polja nije 0 nastavljamo s daljnjim koracima
  - Ako je veličina novog polja 0, trenutno ne postoji aktivni niz te je rezultat 0
4. Iteriramo unazad po vremenskim oznakama početaka dana sve dok se vremenska oznaka početka dana nalazi u polju vremenskih oznaka pokušaja;
  - tijekom svake iteracije povećavamo aktivni niz za 1

Na skoro identičan način računaju se podatci za prikaz vremenske statistike u obliku kalendara i linearnog grafa. Za ispravan prikaz kalendara potreban je samo prvi korak iz prethodne liste. Za ispravan prikaz linearnog grafa, zadnji korak je promijenjen na sljedeći način:

- Iteriramo unazad po vremenskim oznakama početaka dana zadnjih tjedan dana (7 dana)
  - tijekom svake iteracije u polje bodova spremamo broj bodova osvojen tog dana

### 4.2.5 Generiranje lekcija

Aplikacija trenutno ne podržava unaprijed složene lekcije i zadatke, već se potonji uvijek generiraju algoritamski ovisno o tečaju, kategoriji lekcije i trenutnom napretku odnosno statistici. Svaka lekcija (izuzev slovanja) se sastoji od 5 slova ili brojeva. Maksimalan broj određen je empirijski. Ograničenje je uvedeno kako ne bi došlo do preopterećenosti korisnika uslijed istovremenog učenja velikog broja znakova. Izbor znakova vrši se pomoću sljedećeg algoritma:

1. Iz statistike se zbraja broj točnih pokušaja za svaki znak te se znakovi dijele

#### Poglavlje 4. Metodologija

na polje znakova s preko 100 točnih pokušaja (naučeni znakovi) i s manje od 100 točnih pokušaja

2. Nasumično se bira 5 znakova iz grupe znakova koji nemaju preko 100 točnih pokušaja
3. Ako nije moguće izabrati 5 znakova iz grupe znakova koji nemaju preko 100 točnih pokušaja, preostali broj znakova (do 5 sveukupno) se nasumično bira iz polja znakova s preko 100 točnih pokušaja

Navedeni algoritam nije primjenjiv za učenje slovkanja. Slovkanje ne koristi unaprijed definiranih 5 slova ili brojeva, već se u svakom koraku nasumično izabire riječ korištenjem knjižnice za generiranje riječi.

Osim znakova, svakoj lekciji potrebno je generirati broj svakog tipa pojedinog zadatka. Zbroj tipova sveukupno mora biti 15 (maksimalan broj zadataka). Određivanje broja pitanja za pojedini tip zadataka je prikazano u tablici 4.3. Broj zadataka se različito računa u ovisnosti o fizičkoj ili softverskoj blokadi kamere. Također, broj zadataka se različito računa za slovkanje, s obzirom na smanjeni broj tipova zadataka prilagođenih slovkanju.

Tablica 4.3 Određivanje broja pitanja

Tip zadatka	Sa Kamerom	Bez Kamerom
Višestruko spajanje	[3, 4]	[4, 5]
Višestruki izbor (znak)	[4, 5]	[5, 6]
Višestruki izbor (slovo ili broj)	[3, 4]	15 - ostatak $\rightarrow$ [0, 3]
Prepoznavanje s kamerom	15 - ostatak $\rightarrow$ [2, 5]	0
Slovkanje s višestrukim odabirom	8	15
Slovkanje s kamerom	7	0

### **4.2.6 Međuspremnik slika**

Tijekom razvoja i testiranja aplikacije primijećen je problem "uskakanja" (engl. pop-in) slika. Do navedenog problema dolazi jer web preglednik ima sakupljač smeća (engl. garbage collector) koji sve slike koje nisu u uporabi (nemaju HTML img element prisutan na ekranu ili memoriji) briše iz memorije. Ovaj problem je riješen tako da se prije učitavanja lekcije, za svaku potrebnu sliku kreira HTML img element, koji se pohranjuje u posebno kreirano polje za predmemoriju slika. Negativna strana ovakvog pristupa je povećana potrošnja memorije.



# Poglavlje 5

## Rezultati

Aplikacija je postavljena na poslužitelj i dostupna je za testiranje. Poveznica na aplikaciju nalazi se u dodatku A.

Napravljena aplikacija testirana je na uređajima svih veličina i snaga. Generalno opažanje je kako aplikacija nema problema s radom na stolnim računalima te na mobilnim uređajima izdanima tijekom i poslije 2019.-e godine.

U nastavku će biti opisane značajke glavnih dijelova izrađenog sustava te (gdje je moguće) nedostaci istih.

### 5.1 Izgled aplikacije

Po uzoru na postojeće aplikacije te kako bi dizajn bio prilagođen korisniku, za maskotu i glavni motiv aplikacije odabran je pingvin. Sukladno tome, sustav je nazvan PingoLingo.

Za izradu sučelja korištena je Mantine knjižnica gotovih komponenti koja je naknadno oblikovana Cascading Style Sheets (CSS) stilskim jezikom. Sam dizajn i položaj elemenata prvenstveno je orijentiran na uređaje manjih dimenzija, jer su oni puno pristupačniji nego osobna računala.

Početni zaslon aplikacije prikazan je na slici 5.1. Pri pokretanju aplikacije korisnika dočekuje animirana pozadina koja se sastoji od animirane slike pingvina.



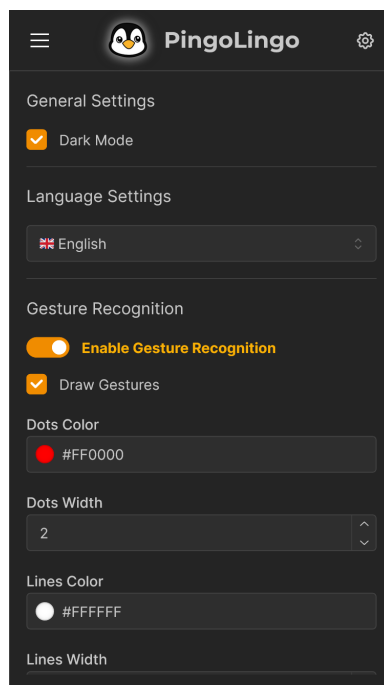
Slika 5.1 Početni zaslon aplikacije

Pozadina se po potrebi zamućuje (e.g. tijekom učenja) kako bi se održala čitkost i preglednost sučelja. Za prikaz slova i brojeva koriste se slike slova i brojeva u motivu pingvina.

S obzirom na izbor pingvina kao temeljnog motiva aplikacije, paleta boja sastoji se primarno od sivih tonova te dominantne narančaste boje. Prednost ovakve palete boja je profesionalni izgled aplikacije, dok je negativna strana težina dizajniranja novih elemenata s obzirom na ograničenu ponudu boja.

## 5.2 Postavke

PingoLingo sustav nudi mogućnost promjene postavki. Postavke se otvaraju pritiskom na krajnje desni gumb na navigacijskoj traci. Pritiskom na gumb, s desne strane aplikacije (preko cijelog ekrana na uređajima manjih dimenzija) otvara se okno s postavkama. Izgled navigacijskog okna postavki prikazan je na slici 5.2.



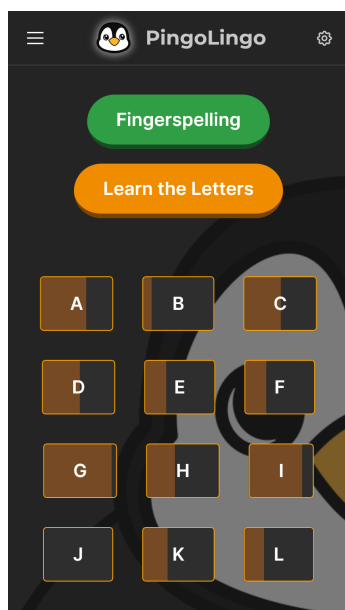
Slika 5.2 Postavke aplikacije

Postavke su podijeljene na 4 glavne kategorije. Unutar prve kategorije nalaze se postavke izgleda. Trenutno je dostupna jedino promjena teme u svijetlu ili tamnu. Unutar druge kategorije nalazi se promjena jezika sučelja na jedan od 3 dostupna jezika: hrvatski, engleski i njemački. Treća kategorija sadrži postavke umjetne inteligencije. Korisniku se daje mogućnost isključivanja zadataka koji koriste kameru i umjetnu inteligenciju te mogućnost promjene izgleda prikaza kostura na prikazu kamere. Zadnja kategorija postavki vezana je uz napredak. Korisniku je omogućena promjena imena (trenutno samo vizualno) te poništavanje napretka. Također, zadnja kategorija sadrži gumb koji korisnika odvodi na pregled statistike, čija je funkcionalnost opisana u odjeljku 5.6.

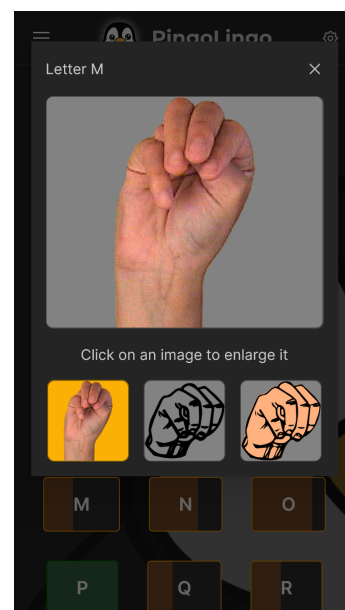
S obzirom na izuzetno fleksibilnu implementaciju postavki, moguća su daljnja proširenja broja postavki. Trenutno glavni nedostatak postavki je nemogućnost biranja modela strojnog učenja, jer tijekom testiranja, ovisno o uvjetima unutar kojih se korisnik nalazi, starije verzije trenutno korištenog modela mogu biti točnije.

## 5.3 Odabir lekcije

Kako je opisano u poglavlju 3, lekcije nije moguće birati, već se one nasumično generiraju. Lekcija se pokreće pritiskom na gumb za učenje. Ovisno o tome nalazi li se korisnik na stranici brojeva ili stranici slova, pokreće se učenje slova ili brojeva. Prikaz izgleda stranice slova prikazan je na slici 5.3a. Osim gumba za pokretanje učenja, na stranici se također nalazi pregled trenutnog napretka učenja za svaki pojedini znak. Potpuno naučeni znakovi označeni su zelenom bojom.



(a) Pregled učenja slova u mobilnom načinu



(b) Otvoren dijaloški okvir pomoći za slovo M

Slika 5.3 Pregled učenja slova

Pritiskom na element napretka, otvara se dijaloški okvir pomoći koji nudi pregled znaka za odabrano slovo ili broj. Skoro svako slovo ili broj imaju 3 različite slike znaka koji ih predstavlja. Redom to su:

- Slika ruke u boji
- Crtež ruke u boji
- Monokromatski crtež ruke

Pritiskom na jednu od ponuđenih podslika, moguće ju je staviti u prvi plan, odnosno, dodatno povećati. Prikaz dijaloškog okvira za pomoć, nalazi se na slici 5.3b.

## 5.4 Lekcija i pitanja

Pokretanjem učenja, odnosno lekcije, izgled navigacijskog okna se mijenja i sadrži gumb za prekidanje lekcije (i povratak na prethodni ekran), traku napretka koja pokazuje postotak riješenosti trenutne lekcije, trenutni broj života te gumb za otvaranje postavki. Traka napretka i broj života mijenjaju boju ovisno o vrijednosti, kako bi se povećala istaknutost elemenata.

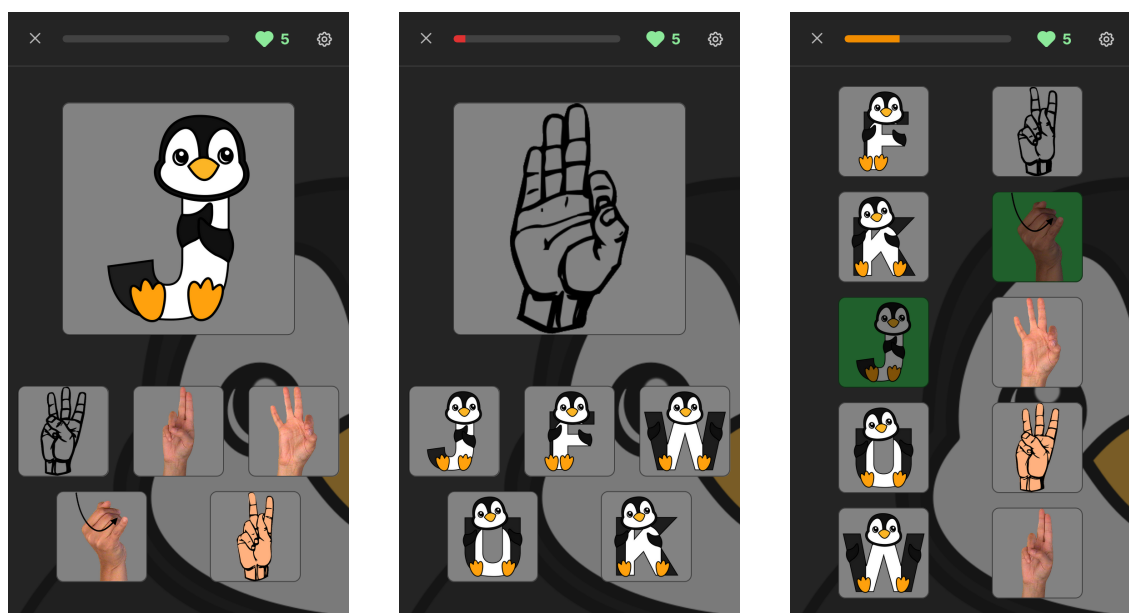
Životi su jedinstveni za trenutnu lekciju te služe kao dodatna inicijativa točnom rješavanju. U pitanjima s više ponuđenih odabira nema dopuštenih grešaka te netočan odgovor oduzima 1 životni bod (srce). Pitanje spajanja znakova sa značenjem omogućava 3 greške prije oduzimanja životnog boda. Dio korisnika sličnih sustava u srodnim aplikacijama ne odobrava značajku života, stoga PingoLingo nudi mogućnost isključivanja oduzimanja života unutar postavki.

Za prikaz slika slova i brojeva koriste se crteži temeljeni na motivu pingvina, dok se znakovi nasumično biraju iz dostupnih slika znakova.

### Pitanje s kamerom

Pitanje s kamerom prikazano je na slici 5.5. Pitanje se sastoji od elementa platna koje iscrtava izlazni signal kamere, trake napretka, statistike detekcije te primjera traženog znaka. Naposljetku se nalazi gumb koji omogućava preskakanje pitanja ako korisnik trenutno ne može koristiti kameru i/ili model nije u stanju ispravno prepoznati pokazane elemente. Statistika detekcije uzima se direktno iz MediaPipe knjižnice te prikazuje uvjerenost u ispravnost trenutne predikcije te o kojoj ruci se radi (lijevoj ili desnoj).

Premda model umjetne inteligencije radi brzinom osvježavanja ekrana (ako je sklopovlje dovoljno jako), brzina validacije je ograničena na provjeravanje svakih 100 milisekundi. Za ispravno rješavanje pitanja, potrebno je pokazati traženi element u



(a) Pitanje s više ponuđenih odabira znaka  
(b) Pitanje s više ponuđenih odabira slova  
(c) Pitanje spajanja znakova sa značenjem

Slika 5.4 Primjeri pitanja

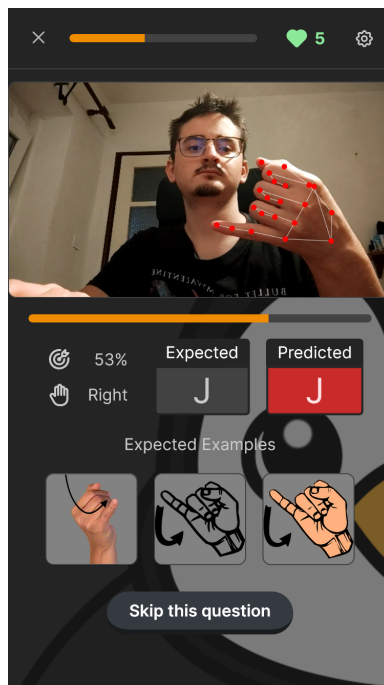
10 validacija u nizu, što odgovara periodu od 1 sekunde. Prednost ovakvog pristupa je što je otporniji na određene rubne slučajeve gdje je na par milisekundi moguća kriva predikcija modela. Dodatna prednost je smanjena količina pozadinskih zadataka koje aplikacija izvršava.

S obzirom na nesavršenosti u modelu za prepoznavanje znakovne abecede, nije implementirana mogućnost krivog rješavanja ovakve vrste pitanja. Jedini mogući ishodi su točno rješenje ili preskakanje pitanja.

### Završetak lekcije

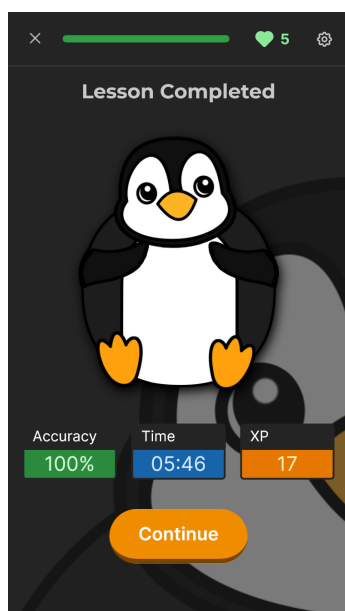
Završetkom lekcije svi pokušaji unutar lekcije spremaju se u lokalni spremnik te se ovisno o uspješnosti prikazuje jedan od 2 ekrana završetka prikazanih na slici 5.6. U slučaju uspješnog završetka prikazuje se animirani Pingvin u boji koji plješće. U slučaju neuspješnog završetka prikazuje se animirani Pingvin u sivim tonovima koji je tužan i hvata se za glavu.

## Poglavlje 5. Rezultati

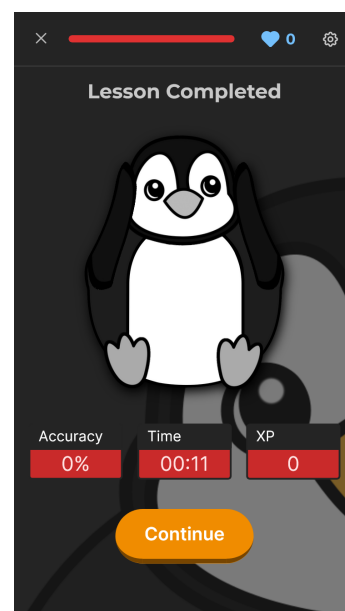


Slika 5.5 Pitanje s kamerom

Animacija služi za razbijanje monotonosti te kao svojevrsna nagrada uspješnog završetka. Nažalost, uz animaciju pozadine, ovo su jedine animacije trenutno dostupne u aplikaciji, jer se njihova implementacija izvodi ručno, pisanjem i optimizacijom istih unutar CSS stilskog jezika. Na uređajima s izuzetno malom veličinom ekrana (i.e. iPhone SE) moguće su greške u prikazu animacija.



(a) Uspješno završena lekcija



(b) Neuspješno završena lekcija

Slika 5.6 Izgled završetka lekcije

## 5.5 Slovkanje

Slovkanje (engl. fingerspelling) je posebna vrsta lekcija koja se otključava kada su svi znakovi abecede naučeni (imaju preko 100 točnih rješavanja). Međutim, ako korisnik želi, slovkanje je moguće prisilno omogućiti unutar postavki. U načinu slovkanja dostupne su samo dvije vrste pitanja prikazane na slici 5.7. Prva vrsta pitanja je slovkanje s višestrukim izborom koje je bazirano na pitanju s višestrukim izborom znakova i pitanju spajanja znakova. Druga vrsta pitanja je prepoznavanje znaka s kamerom te je izvedeno iz običnog pitanja prepoznavanja s kamerom. U pitanju s kamerom dodana je dodatna mogućnost preskakanja određenog slova unutar riječi, po uzoru na Pocket Sign aplikaciju opisanu u odjeljku 2.2.3.

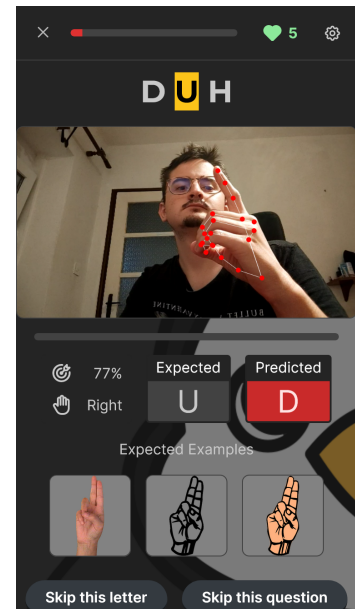
Nasumičan izbor riječi izveden je korištenjem "faker.js" knjižnice. Izabrane riječi mogu biti izuzetno dugačke (duže od 10 slova) što može negativno utjecati na zadovoljstvo učenja te može biti nepregledno na manjim uređajima. Knjižnica podržava određivanje minimalne i maksimalne duljine riječi, ali ta mogućnost trenutno nije



Poglavlje 5. Rezultati



(a) Slovanje s višestrukim izborom



(b) Slovanje s kamerom

Slika 5.7 Izgled zadatka slovanja

implementirana unutar PingoLingo sustava premda postoji priprema za istu.

## 5.6 Statistika

Statistika je implementirana koristeći metode opisane u odjeljku 4.2.4. Pregled statistike zamišljen je kao stranica korisničkog profila. Pristup stranici vrši se pritiskom na gumb koji se nalazi na dnu okna postavki. Takav, relativno skriveni pristup statistici, izabran je kako bi dizajn aplikacije ostao usredotočen na učenje, dok statistika služi kao dodatna značajka. Izgled statistike unutar aplikacije (u punoj veličini) nalazi se na slici 5.8.



Slika 5.8 Statistika unutar aplikacije

Radi preglednosti, statistika je organizirana na 3 osebujna dijela. U prvom se

## *Poglavlje 5. Rezultati*

nalaze opće informacije o korisniku i o korisnikovom napretku, poput broja osvojenih bodova.

U drugom se nalazi statistika u odnosu na vrijeme, u obliku pregleda dana učenja prikazanih na kalendaru te pregleda naučenih bodova u zadnjih 7 dana. Posljednji odjeljak sastoji se od kružnih grafikona koji nude statistički pregled, ovisno o preciznosti, kategoriji lekcije i vrsti pitanja.

Nedostatak pogleda statistike jest njegova statičnost, odnosno nemogućnost dodatne dublje analize primjenom dodatnih filtera. Također, porastom broja pokušaja raste i potrebno vrijeme za računanje statistike te je moguć sporiji rad stranice sa statistikom. U praksi takav slučaj nije uočen, ali je teoretski moguća njegova pojava na slabijim uređajima. Način računanja statistike ovisi o vremenskoj zoni te su u teoriji mogući rubni slučajevi gdje dolazi do greške računanja statistike, jer je korisnik promijenio vremensku zonu.

# Poglavlje 6

## Zaključak

Moderne višeplatformske tehnologije razvoja te napredak na području umjetne inteligencije omogućavaju izradu sustava za učenje znakovnih jezika i abecede. Takvi sustavi za učenje mogli bi učiniti učenje znakovnih jezika pristupačnijim, nego li je moguće tradicionalnim metodama učenja. U ovom radu napravljen je osvrt na trenutno stanje tehnologija ključnih za postizanje potonjeg te je objašnjen proces izrade takvog sustava.

Dostupni skupovi podataka, alati i metode učenja, dostatni su za kreiranje umjetne inteligencije koja sa zadovoljavajućom preciznošću može prepoznati slovanje znakovnom abecedom. Korištenjem umjetne inteligencije, izrađeni sustav omogućuje dodatnu razinu validacije korisnikovog učenja u odnosu na trenutno dostupne sustave. Oslanjanjem na moderne tehnologije razvoja web aplikacija te na arhitekturu koja ne zahtijeva poslužiteljski dio (engl. serverless), izrađeni sustav je izuzetno lako nadograđivati te postaviti na internet.

Neophodno je navesti kako izrađeni sustav nije bez nedostataka te su moguća brojna daljnja poboljšanja. Sama aplikacija ne koristi značajke progresivnih web aplikacija (engl. Progressive Web Application (PWA)) te samim time ne nudi značajke poput notifikacija. Pretvaranjem sustava iz obične web aplikacije u progresivnu web aplikaciju omogućila bi se bolja integracija s mobilnim operacijskim sustavima. Nadalje, drugo interesno područje je poboljšanje strojno učenog modela. To bi se u teoriji moglo postići korištenjem ručno izrađene LTSM neuronske mreže koja bi

## *Poglavlje 6. Zaključak*

koristila videozapise kao ulazni signal a ne statičke slike.

Naposljetku, sam sustav mogao bi se proširiti dodavanjem dodatnih znakovnih abeceda (e.g. britanska znakovna abeceda) te naposljetku proširivanjem na učenje punog znakovnog jezika.

# Literatura

- [1] World federation of the deaf. World Federation of the Deaf. , s Interneta, <https://wfdeaf.org/> , travanj 2024.
- [2] V. Bušelić and K. Župan, “Metode gamifikacije u online sustavima učenja programiranja – osobno iskustvo,” *Polytechnic and design*, vol. 6, no. 4, pp. 245–253, 2018. , s Interneta, <https://doi.org/10.19279/TVZ.PD.2018-6-4-06>
- [3] X. Jiang, J. Rollinson, H. Chen, B. Reuveni, E. Gustafson, L. Plonsky, and B. Pajak. (2021) How well does duolingo teach speaking skills. Northern Arizona Univ. , s Interneta, <https://duolingo-papers.s3.amazonaws.com/reports/duolingo-speaking-whitepaper.pdf> , travanj 2024.
- [4] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, “Machine learning methods for sign language recognition: A critical review and analysis,” *Intelligent Systems with Applications*, vol. 12, p. 200056, 2021.
- [5] R. S. Rosen, “American sign language curricula: A review,” *Sign Language Studies*, vol. 10, no. 3, pp. 348–381, 2010.
- [6] K. Bicknell, C. Brust, and B. Settles, “How duolingo’s ai learns what you need to learn: The language-learning app tries to emulate a great human tutor,” *IEEE Spectrum*, vol. 60, no. 3, pp. 28–33, 2023.
- [7] B. L. Loeding, S. Sarkar, A. Parashar, and A. I. Karshmer, “Progress in automated computer recognition of sign language,” in *International Conference on Computers for Handicapped Persons*. Springer, 2004, pp. 1079–1087.
- [8] K. Nimisha and A. Jacob, “A brief review of the recent trends in sign language recognition,” in *2020 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2020, pp. 186–190.
- [9] A. Ardiansyah, B. Hitoyoshi, M. Halim, N. Hanafiah, and A. Wibisurya, “Systematic literature review: American sign language translator,” *Procedia Computer Science*, vol. 179, pp. 541–549, 2021.

## Bibliografija

- [10] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, “Mediapipe hands: On-device real-time hand tracking,” *arXiv preprint arXiv:2006.10214*, 2020.
- [11] Lingvano. Lingvano GmbH. , s Interneta, <https://www.lingvano.com/asl/> , travanj 2024.
- [12] Asl bloom. SignLab AS. , s Interneta, <https://www.aslbloom.com/> , travanj 2024.
- [13] Pocket sign. MobiReactor. , s Interneta, <https://www.pocketsign.org/> , travanj 2024.
- [14] D. Huynh, L. Zuo, and H. Iida, “An assessment of game elements in language-learning platform duolingo,” in *2018 4th International Conference on Computer and Information Sciences (ICCOINS)*. IEEE, 2018, pp. 1–4.
- [15] Asl alphabet dataset. Kaggle. , s Interneta, <https://www.kaggle.com/dsv/29550> , svibanj 2024.
- [16] Synthetic asl alphabet dataset. Kaggle. , s Interneta, <https://www.kaggle.com/datasets/lexset/synthetic-asl-alphabet> , svibanj 2024.
- [17] Synthetic asl numbers dataset. Kaggle. , s Interneta, <https://www.kaggle.com/datasets/lexset/synthetic-asl-numbers> , svibanj 2024.
- [18] S. Takahashi. (2020, 12) hand-gesture-recognition-using-mediapipe. , s Interneta, <https://github.com/Kazuhito00/hand-gesture-recognition-using-mediapipe> , svibanj 2024.

# Popis slika

2.1	Koraci prepoznavanja znakovnog jezika . . . . .	6
2.2	Izdvojene značajke MediaPipe Hands modela . . . . .	6
2.3	Primjer pitanja unutar Lingvano aplikacije . . . . .	8
2.4	Primjer pitanja unutar ASL Bloom aplikacije . . . . .	9
2.5	Primjer umjetne inteligencije unutar Pocket Sign aplikacije . . . . .	10
3.1	Skica glavnog sučelja . . . . .	13
3.2	Skica sučelja pitanja s više ponuđenih odabira . . . . .	14
3.3	Skica sučelja pitanja s kamerom . . . . .	15
3.4	Skica sučelja pitanja spajanja znakova s značenjem . . . . .	16
3.5	Skica sučelja korisničkog profila i statistike . . . . .	17
4.1	Struktura odbačenog modela . . . . .	20
4.2	Matrica zbunjenosti odbačenog modela . . . . .	20
4.3	Struktura prilagođenog Gesture Recognizer modela . . . . .	21
4.4	Prikaz logike pitanja s kamerom . . . . .	27
5.1	Početni zaslon aplikacije . . . . .	33
5.2	Postavke aplikacije . . . . .	34
5.3	Pregled učenja slova . . . . .	35
5.4	Primjeri pitanja . . . . .	37



*Popis slika*

5.5	Pitanje s kamerom . . . . .	38
5.6	Izgled završetka lekcije . . . . .	39
5.7	Izgled zadatka slovanja . . . . .	40
5.8	Statistika unutar aplikacije . . . . .	41

# Popis tablica

2.1	Usporedba GBA i VBA pristupa . . . . .	5
4.1	Rezultati promjene hiperparametara učenja . . . . .	24
4.2	Skladišta i njihove namjene . . . . .	26
4.3	Određivanje broja pitanja . . . . .	30

# Popis isječaka koda

4.1	Struktura tečaja . . . . .	25
4.2	Struktura znaka . . . . .	25
4.3	Struktura pokušaja rješavanja . . . . .	28

# Pojmovnik

**ASL** American Sign Language. 7

**BSL** British Sign Language. 7

**CNN** Convolution Neural Network. 5, 6

**CSS** Cascading Style Sheets. 32, 38

**GBA** Glove Based Approach. 4, 5, 49

**GIF** Graphics Interchange Format. 14

**LTSM** Long short-term memory. 4, 43

**PWA** Progressive Web Application. 43

**RNN** Recurrent Neural Network. 4

**SPA** Single Page Application. 24

**VBA** Visual Based Approach. 4, 5, 49

**WFD** World Federation of the Deaf. 1

**ÖGS** Österreichische Gebärdensprache. 7

# Sažetak

U radu je istražen napredak u kurikulumu i metodama učenja znakovnog jezika. Uz analizu napretka opisano je trenutno stanje na polju detekcije znakovnog jezika umjetnom inteligencijom. Pregledani su trenutno dostupni komercijalni sustavi za učenje znakovnog jezika. Opisan je proces strojnog učenja modela detekcije znakovnog jezika te utjecaj promjene hiperparametara učenja na preciznost i odziv detekcije. Kroz izrađeni modularni višepatformski sustav za učenje znakovne abecede uz primjenu umjetne inteligencije, dokazano je kako su znanja stečena postojećim istraživanjima dostatna za izradu takvog sustava. Naposljetku, navedena su moguća buduća istraživanja i poboljšanja sustava za učenje.

***Ključne riječi*** — **ASL, Američki znakovni jezik, Znakovni Jezik, Sustav za učenje, MediaPipe, React, Typescript**

## Abstract

The paper explores progress in the curriculum and methods of sign language learning. The current state and progress in sign language detection using artificial intelligence have been examined. Presently available commercial sign language learning systems are reviewed. The machine learning process of the sign language detection model is described, as well as the influence of changing the learning hyperparameters on the precision and recall of the detection. Through the created modular multiplatform system for sign language alphabet learning with the application of artificial intelligence, it has been proven that the knowledge gained from existing research is sufficient for the creation of such a system. Finally, possible future research and improvement of the learning system are listed.

***Keywords*** — **ASL, American Sign Language, Sign Language, Learning System, MediaPipe, React, Typescript**

# Dodatak A

## Poveznice

- Postavljena aplikacija: <https://pingo-lingo.web.app/>
- Izvorni kod: <https://github.com/ijandric97/pingolingo>