

# Mobilna Android aplikacija za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti

---

Fehir, Karla

Undergraduate thesis / Završni rad

2022

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:354296>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-03-29**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**  
**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I**  
**INFORMACIJSKIH TEHNOLOGIJA OSIJEK**

**Sveučilišni studij računarstva**

**MOBILNA ANDROID APLIKACIJA ZA**  
**VIŠEKRITERIJSKO PERSONALIZIRANO**  
**PREPORUČIVANJE FIZIČKIH AKTIVNOSTI**

**Završni rad**

**Karla Fehir**

**Osijek, 2022.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju**

Osijek, 12.09.2022.

Odboru za završne i diplomske ispite

**Prijedlog ocjene završnog rada na preddiplomskom sveučilišnom studiju**

<b>Ime i prezime Pristupnika:</b>	Karla Fehir
<b>Studij, smjer:</b>	Preddiplomski sveučilišni studij Računarstvo
<b>Mat. br. Pristupnika, godina upisa:</b>	R 4341, 22.07.2019.
<b>OIB Pristupnika:</b>	58444996837
<b>Mentor:</b>	Prof.dr.sc. Goran Martinović
<b>Sumentor:</b>	,
<b>Sumentor iz tvrtke:</b>	
<b>Naslov završnog rada:</b>	Mobilna Android aplikacija za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti
<b>Znanstvena grana rada:</b>	<b>Programsko inženjerstvo (zn. polje računarstvo)</b>
<b>Zadatak završnog rada:</b>	U završnom radu potrebno je opisati i analizirati utjecaj pravilnog izbora odgovarajućih fizičkih aktivnosti i vježbi na pretilost, dijabetes, poboljšanje kardiovaskularnog stanja organizma, povećanje mišićne mase, povećanje kvalitete života osoba s tjelesnim oštećenjima i ozljedama i kvalitetu života općenito. Uzimajući u obzir navedene parametre profila korisnika, postojeća slična rješenja, te preporučene
<b>Prijedlog ocjene završnog rada:</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene od strane mentora:</b>	12.09.2022.
<b>Datum potvrde ocjene od strane Odbora:</b>	21.09.2022.
Potvrda mentora o predaji konačne verzije rada:	Mentor elektronički potpisao predaju konačne verzije.
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****IZJAVA O ORIGINALNOSTI RADA**

Osijek, 21.09.2022.

**Ime i prezime studenta:**

Karla Fehir

**Studij:**

Preddiplomski sveučilišni studij Računarstvo

**Mat. br. studenta, godina upisa:**

R 4341, 22.07.2019.

**Turnitin podudaranje [%]:**

12

Ovom izjavom izjavljujem da je rad pod nazivom: **Mobilna Android aplikacija za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti**

izrađen pod vodstvom mentora Prof.dr.sc. Goran Martinović

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.

Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# Sadržaj

1. UVOD .....	1
1.1 Zadatak završnog rada.....	1
2. ZNAČAJ SVAKODNEVNE FIZIČKE AKTIVNOSTI I PRIKAZ STANJA U PODRUČJU..	3
2.1 Definiranje fizičke aktivnosti .....	3
2.2 Važnost fizičke aktivnosti .....	3
2.3 Posljedice zdravstvenih ograničenja i nedostatka fizičke aktivnosti .....	4
2.3.1 Dijabetes.....	4
2.3.2 Pretilost.....	4
2.3.3 Invaliditet .....	5
2.4 Postojeća rješenja za preporučivanje fizičkih aktivnosti.....	5
2.4.1 Home Workout: Health & Fitness.....	5
2.4.2 Senior Fitness .....	6
2.4.3 Injurnymap – Effective exercise therapy .....	7
2.5 Idejno rješenje vlastite aplikacije za preporučivanje fizičkih aktivnosti.....	8
3. MODEL MOBILNE APLIKACIJE ZA PREPORUČIVANJE FIZIČKIH AKTIVNOSTI .....	9
3.1 Funkcionalni zahtjevi mobilne aplikacije .....	9
3.1.1 Prijava korisnika.....	9
3.1.2 Registriranje korisnika .....	9
3.1.3 Unos parametara pri prvoj prijavi .....	9
3.1.4 Izračun indeksa tjelesne mase .....	10
3.1.5 Postupak višekriterijskog stvaranja preporuka fizičke aktivnosti .....	10
3.2.6 Uređivanje korisničkog profila.....	12
3.2.7 Detaljan pregled programa tjelovježbi .....	12
3.2.8 Odjava korisnika .....	12

3.3 Nefunkcionalni zahtjevi mobilne aplikacije.....	12
3.3.1 Performanse.....	12
3.3.2 Prenosivost i podudarnost .....	12
3.3.3 Sigurnost.....	13
3.3.4 Ostali nefunkcionalni zahtjevi.....	13
3.4 Građa mobilne aplikacije .....	13
4. PROGRAMSKO RJEŠENJE APLIKACIJE ZA PREPORUČIVANJE FIZIČKIH AKTIVNOSTI.....	15
4.1 Korišteni alati i tehnologije .....	15
4.1.1 Operacijski sustav Android .....	15
4.1.2 Android Studio .....	15
4.1.3 Programski jezik Kotlin .....	15
4.1.4 Figma.....	16
4.1.5 XML .....	16
4.1.6 Firebase .....	16
4.2. Prikaz programskog rješenja po komponentama .....	17
4.2.1 Registracija i prijava korisnika.....	17
4.2.2 Provjera popunjenosti korisničkih parametara .....	18
4.2.3 Spremanje korisnika u bazu podataka.....	19
4.2.4 Prikaz korisničkih parametara .....	20
4.2.5 Obrada unesenih podataka .....	21
4.2.6 Prikaz preporuka .....	21
4.2.7 Spremanje povijesti vježbanja u bazu podataka.....	23
4.2.8 Prikaz povijesti vježbanja.....	24
4.2.9 Ažuriranje korisničkih parametara .....	25
5. NAČIN KORIŠTENJA I ISPITIVANJE MOBILNE APLIKACIJE S ANALIZOM REZULTATA .....	26

5.1 Korištenje mobilne aplikacije.....	26
5.2 Prikaz rada mobilne aplikacije .....	27
5.2.1 Prikaz prijave korisnika.....	27
5.2.2 Prikaz registracije i unos korisničkih parametara .....	27
5.2.3 Prikaz početnog zaslona .....	28
5.2.4 Prikaz programa tjelovježbe.....	29
5.2.5 Prikaz korisničkog profila .....	30
5.3. Ispitivanje rada mobilne aplikacije .....	31
5.3.1. Korisnički slučaj – Dijabetes.....	31
5.3.2. Korisnički slučaj – Ozlijede koljena .....	32
5.3.3 Korisnički slučaj – Nema zdravstvenih ograničenja .....	33
5.4 Analiza rezultata ispitivanja mob aplikacije .....	34
6. ZAKLJUČAK .....	35
LITERATURA.....	36
POPIS SLIKA .....	38
POPIS TABLICA.....	40
SAŽETAK.....	41
ABSTRACT .....	42
ŽIVOTOPIS .....	43
PRILOZI.....	44

# 1. UVOD

Tjelesna aktivnost bitan je čimbenik za održavanje zdravog i aktivnog života. Svakodnevna fizička aktivnost može uvelike povećati kvalitetu života i raspoloženje. Postoji velik broj načina za postizanje svakodnevne tjelesne aktivnosti sve od šetnje i vožnje bicikla do rekreativnog bavljenja sportom. Svakodnevno kretanje je za neke vrlo lako za postići, dok kod drugih taj cilj može biti teže ostvariv. Uloga računalnih, odnosno informacijskih i komunikacijskih tehnologija u motiviranju na tjelesnu aktivnost, njeno praćenje i analiziranje je vrlo velika. U prvom redu to se odnosi na mogućnosti koje pružaju mobilni uređaji i aplikacije.

Tema ovog završnog rada je osmišljanje, modeliranje i programsko ostvarenje mobilne aplikacije koja ima za cilj olakšati korisnicima pronalaženje idealnih oblika tjelovježbe za ostvarenje njihovog cilja, a to je poboljšanje zdravstvenog stanja i kvalitete života. Pri tom se u obzir uzimaju zdravstveno stanje i ograničenja korisnika, preporuke vježbi temeljene na stručnih znanjima, a preporučivanje fizičkih aktivnosti odnosno tjelovježbe temelji se na višekriterijskom personaliziranom preporučivanju. Ostvarena mobilna aplikacija nudi registriranje korisnika gdje se pri registraciji unose njegovi osobni parametri koji opisuju njegovo fizičko i zdravstveno stanje. Omogućena je prijava korisnika, preporučivanje prikladnog skupa vježbi, praćenje tih odrađenih vježbi i motiviranje korisnika na fizičku aktivnost. Mobilna aplikacija ostvarena je u razvojnoj okolini Android Studio koristeći programski jezik Kotlin, aplikaciju Figma za stvaranje korisničkog sučelja i platformu Firebase za stvaranje baze podataka. Ostvarena mobilna aplikacija prikladno je opisana i analizirana.

U drugom poglavlju opisana je dobrobit svakodnevne fizičke aktivnosti i problemi koji nastaju pri njenom nedostatku. Navedena su i postojeća mobilna rješenja za preporuku fizičke aktivnosti. U trećem poglavlju prikazan je model i građa aplikacije za preporučivanje tjelesne aktivnosti. Opisani su korišteni parametri i sustav stvaranja preporuka. U četvrtom poglavlju definirani su programski alati i tehnologije potrebne za izradu aplikacije, kao i komponente mobilne aplikacije popraćene odgovarajućim programskim kodom. U zadnjem poglavlju prikazan je način rada i upute korištenja mobilne aplikacije, a aplikacija je ispitana i analizirana za odgovarajuće slučajeve korištenja.

## 1.1 Zadatak završnog rada

U završnom radu potrebno je opisati i analizirati utjecaj pravilnog izbora odgovarajućih fizičkih aktivnosti i vježbi na pretilost, dijabetes, poboljšanje kardiovaskularnog stanja organizma, povećanje mišićne mase, povećanje kvalitete života osoba s tjelesnim oštećenjima i ozljedama i



kvalitetu života općenito. Uzimajući u obzir navedene parametre profila korisnika, postojeća slična rješenja, te preporučene vježbe, treba predložiti model i programsku arhitekturu mobilne Android aplikacije s bazom podataka, uključujući funkcionalne i nefunkcionalne zahtjeve. Također, treba predložiti postupak višekriterijskog personaliziranog stvaranja preporuka fizičkih aktivnosti zasnovan na filtriranju sadržaja. Mobilna aplikacija treba omogućiti unos parametara profila korisnika, prikaz i opis vježbi, stvaranje i prikaz preporuka prikladnih vježbi prema profilu i ograničenjima korisnika, te praćenje provedbe navedenih aktivnosti. Mobilnu aplikaciju treba ispitati i analizirati za različite profile korisnika i skupove vježbi.

## **2. ZNAČAJ SVAKODNEVNE FIZIČKE AKTIVNOSTI I PRIKAZ STANJA U PODRUČJU**

U ovom poglavlju definiran je pojam tjelesne aktivnosti i njena važnost u svakodnevnom životu. Analizirane su posljedice nedostatka tjelesne aktivnosti za fizičko i psihološko zdravlje i kvalitetu života.

### **2.1 Definiranje fizičke aktivnosti**

Prvo, važno je definirati tjelesnu aktivnost. Prema [1], Svjetska zdravstvena organizacija (WHO) fizičku aktivnost definira kao “svaki fizički pokret koji zahtijeva utrošak energije aktiviranjem skeletnih mišića”. Prema navedenoj definiciji, fizičku aktivnost predstavlja svaka aktivnost koju obavljamo. Tjelesna aktivnost može se definirati [2] kao svaki fizički pokret uzrokovan djelovanjem mišića koji troši energiju. Redovita i adekvatna tjelesna aktivnost kod odraslih smanjuje rizik od visokog krvnog tlaka, bolesti srca, šećerne bolesti, depresije te općenito poboljšava zdravlje. Osim toga, Svjetska zdravstvena organizacija navodi da je umjerena i energična tjelesna aktivnost važna za poboljšanje zdravlja. Tjelesna aktivnost utječe na zdravlje svih dijelova tijela, posebno mišićno-koštanog, koštanog i kardiovaskularnog sustava.

Tjelesna aktivnost [3] utječe ne samo na zdravlje mišićno-koštanog sustava, već i na bolje mentalno zdravlje. Također, vježbanje nam može pomoći stjecanju osjećaja ponosa kao kada dosegneмо veliko postignuće. Dokazano je da redovita fizička aktivnost pomaže poboljšanju sna i koncentracije. Dokazano je da je sjedilački način života [4] faktor rizika za razvoj mnogih kroničnih bolesti, uključujući kardiovaskularne bolesti. S druge strane, aktivan život povezan je s mnogim društvenim i psihološkim dobrobitima, a povezan je i s produljenim životnim vijekom. Osobe koje svakodnevno održavaju aktivan život pokazuju tjelesno i psihičko zadovoljstvo te bolju kvalitetu života.

### **2.2 Važnost fizičke aktivnosti**

Prema [5], neke od dobrobiti redovite tjelesne aktivnosti uključuju poboljšanje zdravlja, jačanje kostiju i mišića, postizanje zdrave težine, bolju ravnotežu, snagu i fleksibilnost, povećanje energije i smanjenje stresa. Tjelesna aktivnost može poboljšati mentalno zdravlje te pomoći osobama koje boluju od depresije i tjeskobe i zaustaviti razvoj takvih problema. Nakon što osoba postane tjelesno aktivna, počinje razmišljati o drugim gledištima u životu, poput zdrave prehrane i boljeg načina života.

## **2.3 Posljedice zdravstvenih ograničenja i nedostatka fizičke aktivnosti**

Svijet [6] je više nego ikad zabrinut zbog pojave kroničnih nezaraznih bolesti, uključujući kardiovaskularne bolesti, dijabetes, pretilost i mentalnih bolesti. Dok su ljudi većinom svjesni posljedica velikog unosa hrane i nedostatka tjelesne aktivnosti, posljedice sjedilačkog načina života često se zaboravljaju. Jedan od razloga manje tjelesne aktivnosti je i korištenje modernih tehnologija koje olakšavaju život. Manje se kreće, troši se manje energije, vrijeme se provodi ispred računala, a ostale aktivnosti manje su zahtjevne od prethodnih generacija. Istraživanja su otkrila vezu između tjelesne neaktivnosti i psihofizioloških poremećaja (anksioznost i depresija). Anketama o životnim navikama ispitanici su potvrdili da vježbanje, posebice kardiovaskularno, može smanjiti stres i umor te poboljšati cjelokupnu psihofiziološku izvedbu.

### **2.3.1 Dijabetes**

Dijabetes [7] je sedmi vodeći uzrok smrtnosti i jedan od vodećih uzroka morbiditeta. Tjelesno vježbanje pomaže u prevenciji ali i liječenju dijabetesa. Ako se osobe s većim rizikom za razvoj dijabetesa, poput onih s predijabetesom, uključe ranije u strukturirane programe vježbanja, značajno će smanjiti rizik za progresiju u šećernu bolest. Dijabetes tipa 1 je autoimuna bolest koja rezultira nedostatkom inzulina. Dijabetes tipa 2 ima jaču genetsku komponentu koja uključuje kombinaciju inzulinske rezistencije i poremećenog lučenja inzulina. Ova dva nedostatka najlakše se izražavaju u okruženju pretilosti koja je rezultat nezdrave prehrane i sjedilačkog načina života. Iako tjelovježba može imati pozitivnu ulogu u potencijalnom sprječavanju kardiovaskularnih komplikacija ili čak odgađanju pojave dijabetesa tipa 2, ona može pogoršati komplikacije ili izazvati posljedice zbog postojećih komplikacija. Iz tog razloga je bitno stvoriti personaliziranu preporuku za svakog pojedinca. S obzirom na visoku stopu kardiovaskularnih bolesti u bolesnika s dijabetesom tipa 2, programi vježbanja posebno su važni za smanjenje čimbenika rizika za vaskularne komplikacije. Stalna tjelesna aktivnost predložena je kao sredstvo za odgađanje ili sprječavanje pojave dijabetesa tipa 2. Pojedinci koji održavaju tjelesno aktivan način života imaju smanjen rizik od poremećene tolerancije glukoze i dijabetesa tipa 2.

### **2.3.2 Pretilost**

Prema [8] pretilost se definira kao BMI od 31 i više, a teška pretilost kao BMI od 35 i više. Postoje važne razlike između pretilih i teško pretilih pacijenata s obzirom na toleranciju na tjelovježbu. Većina pretilih može se baviti aktivnostima niskog intenziteta kao što je hodanje i u nekoliko slučajeva brzo hodanje. S druge strane, većina teško pretilih osoba, osobito onih s BMI od 40 i iznad, imaju velike poteškoće u obavljanju čak i najjednostavnije vrste tjelesne aktivnosti. Vježba velikog volumena i niskog intenziteta, koja se izvodi gotovo svakodnevno i generira značajno

povećanje dnevne potrošnje energije poboljšava osjetljivost na inzulin i profil lipo proteina u plazmi, kao i sniženi krvni tlak. Iako su ove korisne metaboličke promjene u dobroj korelaciji s gubitkom tjelesne težine i masnoće, trening izdržljivosti također može potaknuti neka metabolička poboljšanja u odsutnosti gubitka težine. Nema sumnje da se većina ovih blagotvornih učinaka vježbanja mogu postići kratkotrajnim programom vježbanja većeg intenziteta. Vježbanje niskog intenziteta s produljenom izdržljivošću može predstavljati najprikladniji oblik tjelovježbe koji se preporučuje općoj populaciji, osobito pretilim pacijentima te za prevenciju dijabetesa i kardiovaskularnih bolesti.

### **2.3.3 Invaliditet**

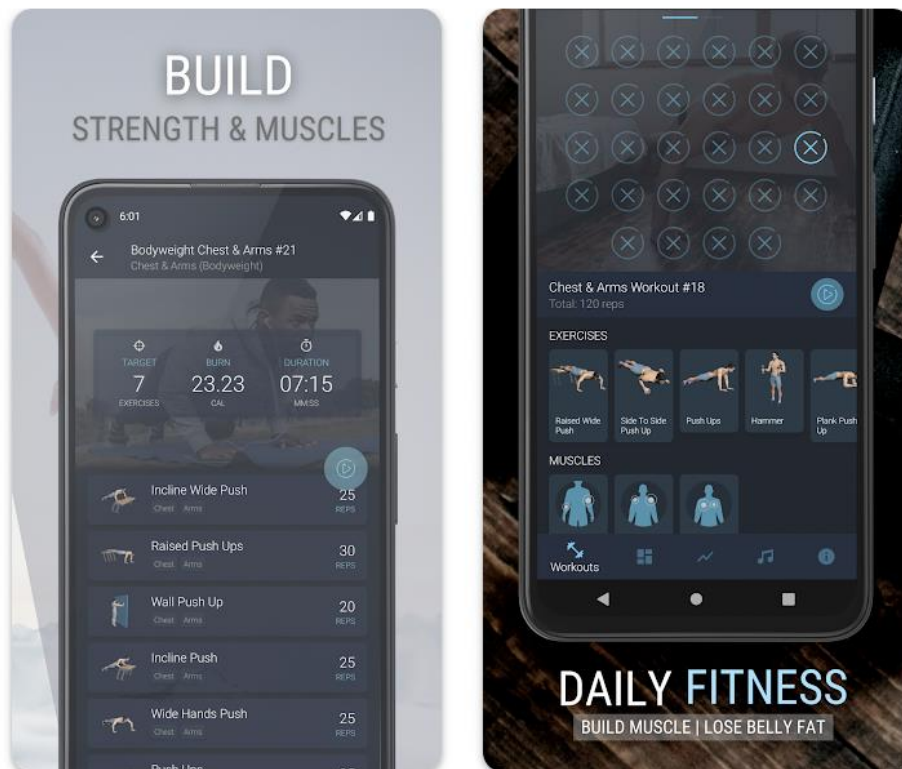
Prema [9] godine 2011. Svjetska zdravstvena organizacija procijenila je da otprilike 15% svjetske populacije živi s nekim oblikom invaliditeta. Dakle, što tjelesna aktivnost znači u životima osoba s invaliditetom? Ako je invaliditet nastao tek kasnije u životu, tjelesna aktivnost se često uvodi u život takvih osoba u svrhu oporavka. Nažalost, tu tjelesna aktivnost prestaje u većini slučajeva. Vježbanje ima mnoge prednosti i svakako se treba poticati. Pozitivni učinci tjelesne aktivnosti počinje s pretpostavkom da nije sama tjelesna aktivnost aspekt koji poboljšava naš pogled na život i mentalno zdravlje, već da nam komunikacija s drugim ljudima tokom vježbanja donose zadovoljstvo. Drugo objašnjenje za pozitivne učinke tjelesne aktivnosti na naše mentalno zdravlje temelji se na pozitivnim mislima koji se javljaju nakon odrađene fizičke aktivnosti. Kada savladamo fizički zahtjevne zadatke, vjerojatno ćemo iskusiti osjećaj samoučinkovitosti. Tjelesna aktivnost prvenstveno se odnosi na izgradnju snage. Osim toga, ljudi koji sustavno vježbaju imaju veću vjerojatnost da će razviti mentalno zdravlje od onih koji to ne rade. Mentalno zdravlje znači snagu, snagu, pokretljivost, fleksibilnost, prilagodljivost, otpornost i izdržljivost. Mentalno zdravlje podrazumijeva psihičko zdravlje, funkcioniranje, pripremu i kontinuirani rad na sebi s ciljem poboljšanja vlastitog učinka.

## **2.4 Postojeća rješenja za preporučivanje fizičkih aktivnosti**

Na tržištu već postoje razvijena rješenja za praćenje fizičke aktivnosti. U nastavku su opisana tri mobilna rješenja.

### **2.4.1 Home Workout: Health & Fitness**

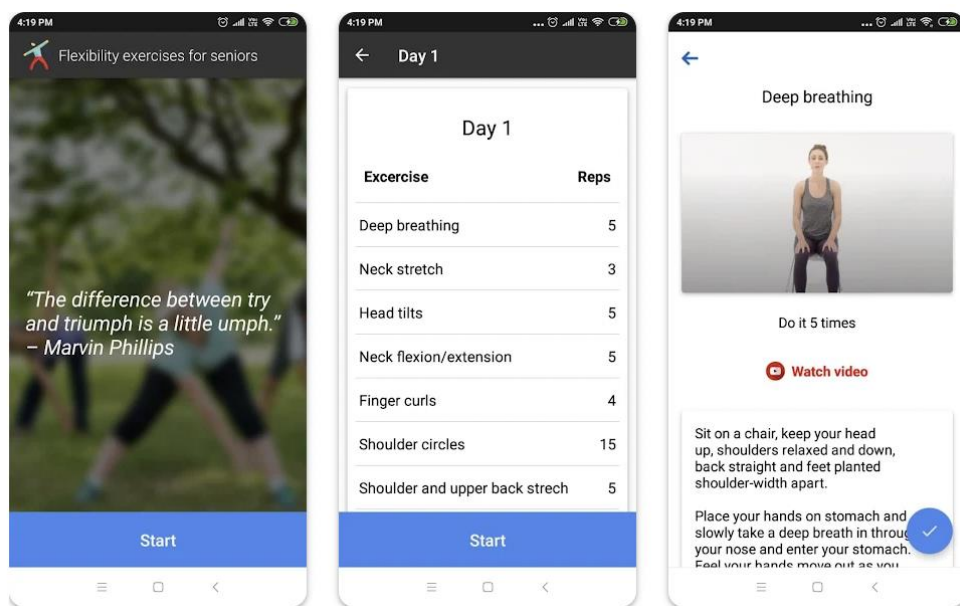
Aplikacija [10] za fitnes, bodybuilding i mršavljenje koja pruža svakodnevni trening za sve glavne skupine mišića. Idealna za kućne treninge, ali se može koristiti i u teretani za treniranje mišića i kondiciju. Izazovni treninzi osmišljeni su za gubitak masti i povećanje mišićne mase. Nudi korisniku mogućnost odabira idealnog plana vježbanja s obzirom na cilj i praćenje napretka. Svaka vježba u programu sadrži kratki opis i ilustraciju kao upute izvođenja.



**Slika 2.1** Sučelje aplikacije Home Workout

#### **2.4.2 Senior Fitness**

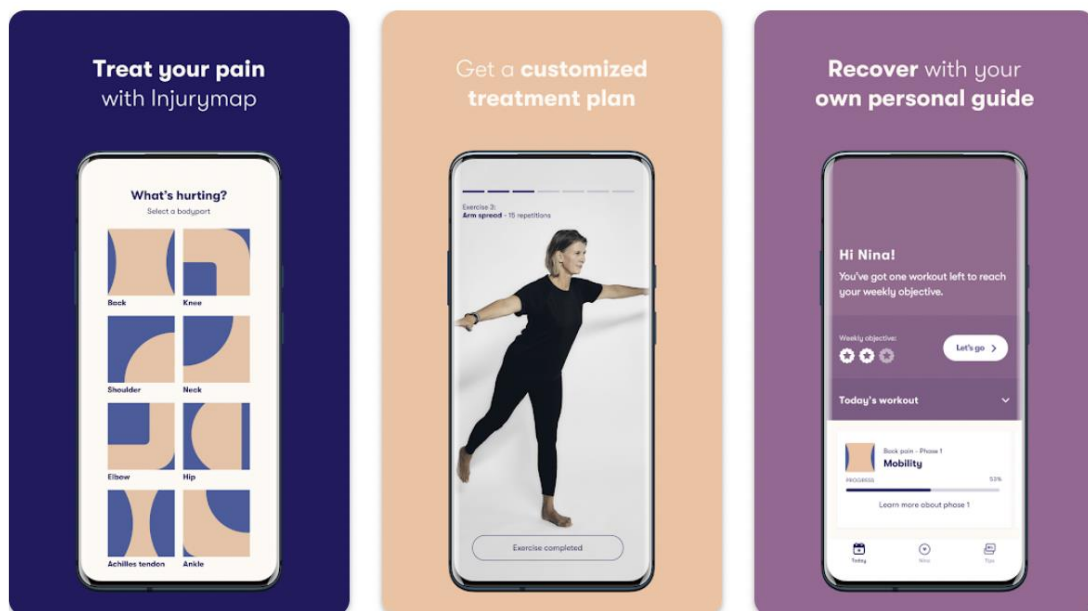
Senior Fitness [11] sadrži dnevne rutine vježbanja za starije osobe u svrhu povećanja fleksibilnosti. Sve vježbe se mogu izvoditi kod kuće, bez odlaska u teretanu. Ova je aplikacija namijenjena onima koji ne mogu ići u teretanu ili raditi teške vježbe, a žele postići cilj redovne tjelesne aktivnosti. Prikazuje vrlo korisne tekstualne i video upute grupirane po danima u tjednu. Također sadrži motivacijske citate kako bi potakla korisnike na što češću tjelesnu aktivnost.



Slika 2.2 Sučelje aplikacije Senior Fitness

### 2.4.3 Injurymap – Effective exercise therapy

Pametni algoritam treninga Injurymapa [12] vodi korisnike kroz liječenje kontinuiranim prilagođavanjem vježbi kako bi upravljali specifičnom boli. Nudi brojne vježbe koje se mogu izvoditi kod kuće fokusirane na odabirnu tjelesnu ozljedu. Također sadrži motivacijske citate kako bi potakla korisnike na što češću tjelesnu aktivnost. Nudi korisniku odabir ozljede te prema tom odabiru koristi sustav preporuka kako bi pronašao idealan program tjelovježbe za osobu.



Slika 2.3 Sučelje aplikacije Injurymap

## **2.5 Idejno rješenje vlastite aplikacije za preporučivanje fizičkih aktivnosti**

Komponente opisane u prethodnom dijelu poglavlja i navedena mobilna Android rješenja bile su inspiracija za razvoj mobilne Android aplikacije za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti. Prvo navedeno mobilno rješenje korisniku preporuča program vježbanja na temelju odabrane mišićne skupine. Vrlo slično tome, drugo mobilno rješenje nudi program fleksibilnosti za starije uzraste koji žele održati određenu razinu aktivnosti. Zadnje mobilno rješenje koristi sustav preporuka koji na temelju odabrane ozlijede korisniku nudi vježbe fleksibilnosti za to područje tijela. Spoj navedenih mobilnih rješenja bile su motivacija za razvoj mobilne aplikacije za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti. Korisnik će imati mogućnosti registrirati tj. prijaviti se u sustav i unijeti svoje parametre. Na temelju unesenih parametara predlažu mu se preporuke programa tjelovježbe koji sadrže detalje o trajanju vježbe, broju potrošenih kalorija i video u kojem se detaljno demonstriraju vježbe. Korisnik može pratiti povijest odrađenih tjelovježbi i broj potrošenih kalorija po čemu se razlikuje od prijašnje navedenih mobilnih rješenja. Detaljan opis modela aplikacije za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti opisan je u poglavlju 3.

### **3. MODEL MOBILNE APLIKACIJE ZA PREPORUČIVANJE FIZIČKIH AKTIVNOSTI**

Ovo poglavlje daje opis funkcionalnih i nefunkcionalnih zahtjeva na mobilnu aplikaciju, opis modela aplikacije za preporučivanje fizičkih aktivnosti i njenu građu.

#### **3.1 Funkcionalni zahtjevi mobilne aplikacije**

##### **3.1.1 Prijava korisnika**

Prilikom otvaranja mobilne aplikacije, korisnik će vidjeti zaslon za prijavu gdje se nalaze polja za unos e-mail adrese i lozinke. Ispod polja za unos nalazi se gumb za prijavu. Pritiskom na gumb za prijavu, ukoliko je unesena ispravna i postojeća e-mail adresa i lozinka, korisnik se prijavljuje na svoj račun te odlazi na početni zaslon. Ukoliko korisnik nema registriran korisnički račun, na zaslonu se također nalazi gumb koji korisnika vodi na zaslon za registriranje.

##### **3.1.2 Registriranje korisnika**

Na zaslonu za registraciju nalaze se polja za unos e-mail adrese i lozinke kojim se korisnik registrira u sustav. Registracija se izvršava pomoću Firebase Auth koji štiti korisničku lozinku dopuštajući stvaranje jedinstvenog računa po jednoj e-mail adresi. Ispod polja za unos adrese e-pošte i lozinke nalaze se dodatna polja za unos korisničkih parametara. Na dnu zaslona ispod svih polja, nalazi se gumb za registraciju koji registrira korisnika u sustav. Na zaslonu se također nalazi gumb koji korisnika vodi na zaslon za prijavu ukoliko ima registriran korisnički račun.

##### **3.1.3 Unos parametara pri prvoj prijavi**

Korisnik upisuje svoje podatke ime, prezime, spol i dob. Ali ti podaci nisu toliko važni za sustav preporuka i odabira prikladnih programa tjelovježbe. Najvažniji parametri za višekriterijsko preporučivanje su visina, težina i zdravstvena ograničenja korisnika. Iz unesenih podataka težine i visine izračunava se indeks tjelesne mase (BMI). Ovo je jedan od važnih čimbenika pri odabiru vježbi koje su najprikladnije za korisnika. Iz padajućeg izbornika korisnik izabire ima li zdravstvenih ograničenja kao što su dijabetes i tjelesne ozljede. Prema tim unesenim podacima se preporučuje personaliziran program tjelovježbe za korisnika. U tablici 3.1 su prikazani parametri koje korisnik unosi pri prvoj prijavi, osim indeksa tjelesne mase koji se računa na temelju korisnikove visine i težine.



**Tablica 3.1** Prikaz korisničkih parametara

Parametar	Tip varijable
Ime	String
Visina	Double
Težina	Double
Starost	Integer
Spol	String
Zdravstvena ograničenja	String
Indeks tjelesne mase	Double

### 3.1.4 Izračun indeksa tjelesne mase

Među najvažnijim podacima za filtriranje odgovarajućih programa tjelovježbi su visina, težina i zdravstvena ograničenja korisnika. Indeks tjelesne mase osobe određuje se na temelju visine i težine. Pri popunjavanju parametara, visina se unosi u centimetrima, ali se to korigira u izračunu. Indeks tjelesne mase izračunava se kao:

$$BMI = \frac{\text{masa}}{\text{visina}^2} \quad (3-1)$$

### 3.1.5 Postupak višekriterijskog stvaranja preporuka fizičke aktivnosti

Sustav stvaranja preporuka [13] uglavnom prolazi kroz tri faze, a to su modeliranje, predviđanje i preporučivanje, što je opisano u nastavku.

Najčešće korištene tehnike preporuka [14] su filtriranje temeljeno na sadržaju, suradničko filtriranje i hibridno filtriranje. Jedno kriterijsko stvaranje preporuka moglo bi proizvesti preporuke koje možda neće zadovoljiti sve potrebe korisnika.

Dokazano je [15] da je sustav stvaranja preporuka presudan u mnogim područjima kao što su kupovina (Amazon), glazba (Spotify), filmovi (Netflix) i ostali. Model sustava preporuka generalno se sastoji od dva skupa i funkcije korisnosti. Skup „Users“ predstavlja sve korisnike, skup „Items“ predstavlja sve što korisniku može biti preporučeno te funkcija korisnosti predstavlja funkciju koja pronalazi personaliziranu preporuku za korisnika.

**Faza modeliranja** usmjerena je na pripremu podataka koje ćemo koristiti u sljedeće dvije faze. U ovoj fazi postoje tri dijela. Prva dio je matrica u kojoj redci predstavljaju korisnike te stupce predstavljaju stvari tj. programi tjelovježbe koji mogu biti preporučeni korisniku. Drugi dio je korisnički profil i parametri koji opisuju korisnika. Treći dio je profil programa koji predstavlja sve stavke, u ovom slučaju specifične tjelovježbe koje se nalaze u programu.

**Faza predviđanja** ima za cilj predvidjeti nevidljive tj. nepoznate stavke za specifičnog korisnika kroz funkciju korisnosti s podacima iz prethodne faze.

**Faza preporuke** je proširenje faze predviđanja gdje se postupkom filtriranja preporučuje najprikladnije rješenje za personalizirano za korisnika.

**Višekriterijsko stvaranje preporuka** [14] svrstava se u hibridne metode stvaranja preporuka te daje korisnicima priliku da specificiraju svoje preferencije na temelju više atributa. Osim toga, višekriterijsko stvaranje preporuka pruža dodatne informacije o preferencijama korisnika u vezi nekoliko važnih aspekata. Postupak višekriterijskog stvaranja preporuka primjeren je za probleme kod kojih su ciljevi vrlo složeni.

Za mobilnu aplikaciju preporuke fizičkih aktivnosti korišten je sustav višekriterijskog stvaranja preporuka. Korisniku se na temelju unesenih korisničkih parametara prema tablici 3.1 stvara personalizirani program tjelovježbe. Glavni parametri za višekriterijsko stvaranje preporuka su indeks tjelesne mase i zdravstveno ograničenje. U tablici 3.2 definirani su mogući programi tjelovježbe te oznake za svaki program. Svaki od navedenih programa tjelovježbe u sebi sadrži skup vježbi preporučenog za tog korisnika. Najvažniji parametar za filtriranje programa tjelovježbe je zdravstveno ograničenje korisnika. Prema odabranom zdravstvenom ograničenju korisniku se preporučuje program tjelovježbe prema tablici 3.3. Ukoliko korisnik kao zdravstveno ograničenje navede ozljedu leđa, preporučuju mu se programi 3,8, i 9 prikazani u tablici 3.2.

**Tablica 3.2** Opis i oznaka programa vježbe

Opis programa vježbe	Oznaka programa vježbe
Program za gornji dio tijela	1.
Program za donji dio tijela	2.
Program za kardio	3.
Program za dijabetes	4.
Program za ramena	5.
Program za leđa i biceps	6.
Program za prsa i triceps	7.
Program za kvadriceps	8.
Program za zadnju ložu	9.
Program za istezanje	10.

**Tablica 3.3** Preporuka programa na temelju ozljede

Opis zdravstvenog ograničenja	Oznaka programa vježbe
Nema zdravstvenog ograničenja	1, 2, 3
Dijabetes	3, 4, 10
Invaliditet	5, 6, 7
Ozljeda koljena	5, 6, 7
Ozljeda leđa	3, 8, 9

### **3.2.6 Uređivanje korisničkog profila**

Korisnik ima mogućnost izmjene unesenih parametara ukoliko su krivo uneseni ili radi ažuriranja napretka (gubitak ili povećanje trenutne kilaže). Korisnik na svome profilu vidi svoje trenutne podatke, te klikom na gumb „Uredi“ ih može promijeniti.

### **3.2.7 Detaljan pregled programa tjelovježbi**

Na početnoj stranici, nakon unosa korisničkih parametara, nalazi se korisnikov indeks tjelesne mase i preporučeni programi tjelovježbe. Klikom na program korisnik ima pregled svih vježbi u tom programu kao i njihov detaljan opis izvođenja, slika i broj ponavljanja. Na vrhu stranice nalazi se omjerno vrijeme potrebno za izvođenje cijelog programa tjelovježbe.

### **3.2.8 Odjava korisnika**

Mobila aplikacija korisniku nudi mogućnost odjave. Podatci koje je korisnik unio ostaju zaštićeni u bazi podataka. Korisnik se u bilo kojem trenutku može ponovno prijaviti u mobilnu aplikaciju.

## **3.3 Nefunkcionalni zahtjevi mobilne aplikacije**

Nefunkcionalni zahtjevi opisuju kako će programska podrška raditi. Oni proizlaze iz funkcionalnih zahtjeva, tako da je najbolje pokazati koji su zahtjevi potrebni u smislu funkcionalnosti. Neki od parametara koji ukazuju na to kakav bi sustav trebao biti su performanse, pouzdanost, održivost te sigurnost. U predloženom sustavu bit će potrebno usredotočiti se na nefunkcionalne zahtjeve koji su najnužniji u smislu funkcionalne potrebe. Neispunjavanje nefunkcionalnih zahtjeva može dovesti do velikih poteškoća jer ih se ne može izbjeći, zbog čega su kritičniji od funkcionalnih zahtjeva.

### **3.3.1 Performanse**

Performanse aplikacije definiraju se kao potreban vremenski interval da aplikacija izvrši zadatak. Prvo je vrijeme "do prvog zaslona", a odnosi se na vrijeme sve od pokretanja mobilne aplikacije do prikazivanja prvog zaslona korisničkog sučelja. Unutar nekoliko sekundi, nakon što se korisnik prijavi i unese korisničke parametre definirane u tablici 3.1, aplikacija mora odabrati personalizirani program tjelovježbe i preusmjeriti korisnika na ekran s popisom vježbi.

### **3.3.2 Prenosivost i podudarnost**

Prenosivost i podudarnost aplikacije odnosi se na kompatibilnost mobilne aplikacije s drugim uređajima. Prema [16] temeljna komponenta prenosivosti programa je Application Programming Interface (API). API se koristi za strukturirani prijenos podataka između softverskih aplikacija.

### **3.3.3 Sigurnost**

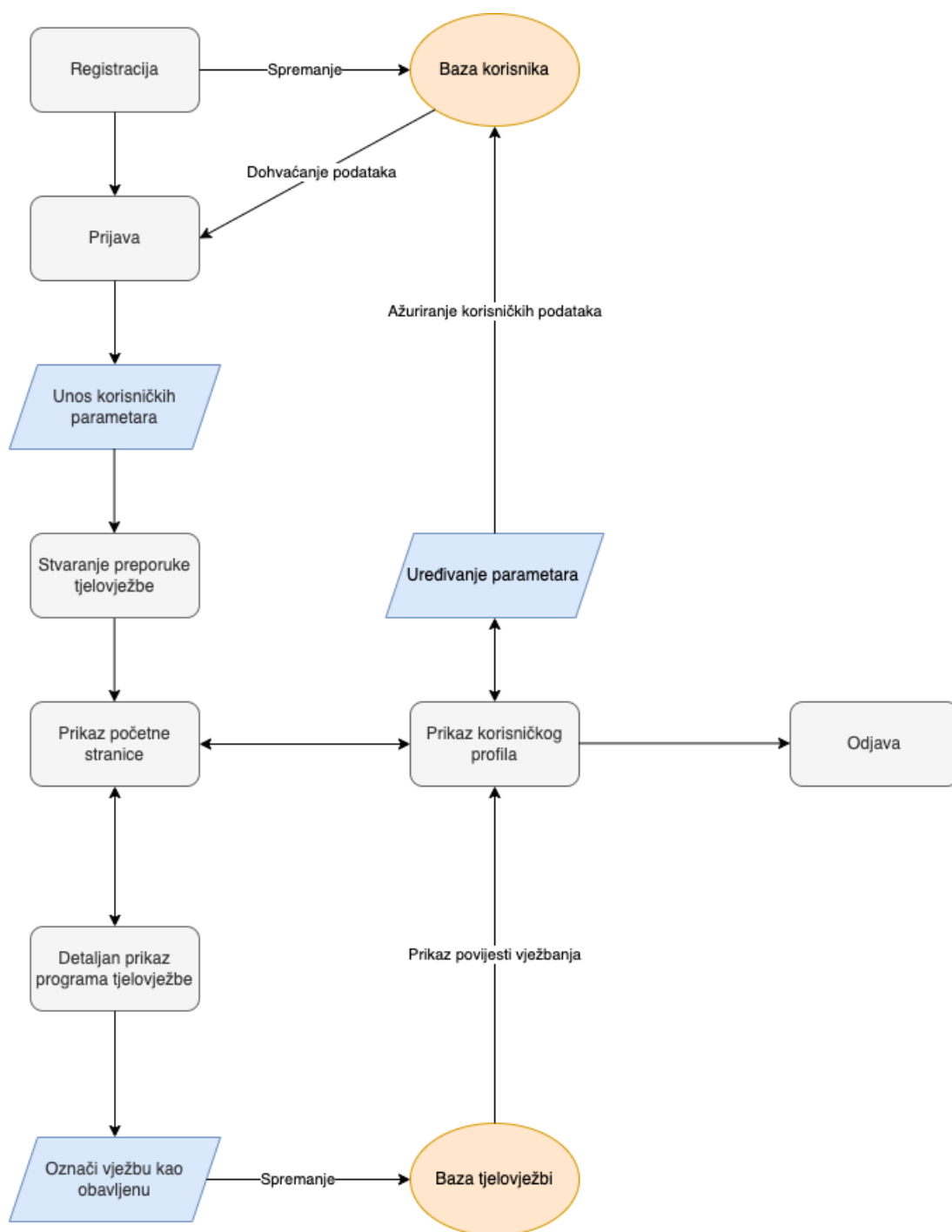
Treba osigurati sigurnost aplikacije kako bi se očuvao njezin integritet od svih vrsta povreda, bile to unutarnje ili vanjske. Time osigurava svoju pouzdanost da će obraditi postavljene zahtjeve.

### **3.3.4 Ostali nefunkcionalni zahtjevi**

Osim performansi, prenosivosti, kompatibilnosti i sigurnosti mobilne Android aplikacije, važno je navesti i ostale nefunkcionalne zahtjeve aplikacije. Mogućnost korištenja aplikacije olakšana je signalizacijom ugrađenom u aplikaciju. Za svaki unos je napisano ono što se od korisnika traži, a ikone označavaju elemente kao što su visina i težina. Time je aplikacija dostupna svim korisnicima.

## **3.4 Građa mobilne aplikacije**

Arhitektura mobilne aplikacije za preporuku fizičkih aktivnosti prikazana je dijagramom na slici 3.2. Odvijanje tijeka mobilne aplikacije detaljno je opisano u funkcionalnim zahtjevima mobilne aplikacije. Kao što je vidljivo u dijagramu mobilne aplikacije, potrebne su dvije baze podataka. Kao bazu podataka koristit će se Firebase. U prvoj bazi podataka spremljeni su korisnički podatci tj. parametri koje je korisnik unio pri prijavi. Također je moguće naknadno uređivanje parametara gdje se ažuriraju i podatci u bazi podataka. U drugoj bazi podataka pohranjeni su programi tjelovježbi koje je korisnik označio kao obavljene. Vježbe iz druge baze podataka prikazane su na korisničkom profilu pod „Povijest vježbanja“.



**Slika 3.1** Dijagram tijeka mobilne aplikacije za preporuku fizičke aktivnosti

## **4. PROGRAMSKO RJEŠENJE APLIKACIJE ZA PREPORUČIVANJE FIZIČKIH AKTIVNOSTI**

### **4.1 Korišteni alati i tehnologije**

Aplikacija za preporučivanje fizičkih aktivnosti dizajnirana je u alatu Figma. Nakon toga izrađena je u razvojnom okruženju Android Studio. Napisana je u programskom jeziku Kotlin, a za korisničko sučelje koristi se jezik XML. Za sustav baze podataka koristi se Firebase.

#### **4.1.1 Operacijski sustav Android**

Android [17] radi na jezgri Linuxa. Android aplikacije se razvijaju pomoću programskih jezika Java i Kotlin. Google ima vlastiti SDK koji omogućuje programskim jezicima upravljanje uređajima. Razvoj mobilnih aplikacija za Android pruža fleksibilnu platformu za programere gdje mogu koristiti i Java IDE i Android java biblioteke. Danas je Android operacijski sustav važna platforma koja pruža dinamičan način razvoja inovativnih aplikacija. Operacijski sustav Android stekao je popularnost među programerima zbog svoje prilagodljive prirode. Vrlo je učinkovito izgraditi aplikaciju na jednoj platformi i istovremeno je implementirati na nekoliko različitih platformi bez brige o promjenama koje treba napraviti. Sigurnost je glavna briga Android uređaja. Ne dopušta vanjskim aplikacijama da mijenjaju ili modificiraju instalirane datoteke. Omogućuje prilagodbu dopuštenja koja specificira aplikacije za korištenje sklopovskih i programskih resursa uređaja.

#### **4.1.2 Android Studio**

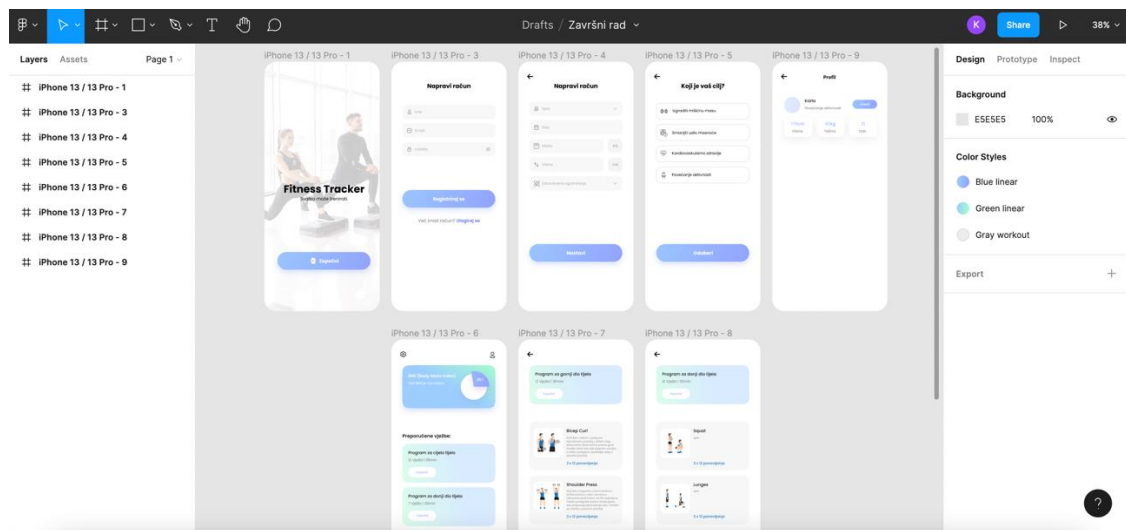
Android Studio [19] službeno je razvojno okruženje utemeljeno na IntelliJ IDEA za razvoj Android aplikacija. Uz IntelliJ uređivač koda i alate za programere, nudi i druge značajke koje povećavaju proizvodnost. Neke od ovih značajki su fleksibilan sustav temeljen na Gradleu, efikasan emulator, jedno okruženje u kojem se mogu razvijati mobilne aplikacije za sve uređaje, primijena promjena na pokrenute aplikacije bez ponovnog pokretanja aplikacije, GitHub integracija te opsežni alati za ispitivanje rada aplikacije. Najčešće korišteni jezik pri izradi Android aplikacija je Java, no sve se više koristi Kotlin.

#### **4.1.3 Programski jezik Kotlin**

Kotlin [20] je objektno orijentirani programski jezik otvorenog koda. Slične je sintakse kao što su C # i Java. Google ga je prihvatio kao glavni programski jezik za Android mobilne aplikacije. Od 2019. posredni jezik je Java.

#### 4.1.4 Figma

Figma [21] je alat za izradu prototipa koji se prvenstveno temelji na webu, s dodatnim izvan mrežnim značajkama koje omogućuju desktop aplikacije za macOS i Windows. Mobilna aplikacija Figma za Android i iOS omogućuje pregled i interakciju s Figma prototipima na mobilnim uređajima u stvarnom vremenu. Skup značajki Figma usredotočen je na korištenje u korisničkom sučelju i dizajnu korisničkog iskustva, s naglaskom na suradnji u stvarnom vremenu. Ima mogućnost izvoza dijelova dizajna kao što su slike, boje i fontovi u Android Studio te olakšava izradu korisničkog sučelja u XML opisnom jeziku. Na slici 4.1 vidljivo je korisničko sučelje alata Figma.



Slika 4.1 Korisničko sučelje alata Figma

#### 4.1.5 XML

Extensible Markup Language (XML) je označni jezik i format datoteke za pohranu, prijenos i rekonstrukciju proizvoljnih podataka. Predstavlja skup pravila za kodiranje dokumenata u format koji je čitljiv i za ljude i za stroj.

#### 4.1.6 Firebase

Firebase [22] omogućuje razvoj mobilnih i web stranica. Kako bismo spremili potrebne podatke za našu vezu s kreiranim korisnikom, implementirana je baza podataka. Vrlo je jednostavna za korištenje i omogućuje praćenje podataka u stvarnom vremenu. Dohvaćanje i spremanje podataka odvija se brzo, što olakšava stvaranje aplikacija koje moraju komunicirati s bazama podataka u velikom broju.

## 4.2. Prikaz programskog rješenja po komponentama

U ovom poglavlju nalaze se komponente ove mobilne Android aplikacije. Opisan je rad svake komponente i prikazani ključni dijelovi koda od koje se je komponenta sastavljena.

### 4.2.1 Registracija i prijava korisnika

Za registraciju korisnika u sustav korišten je Firebase Auth i metoda „CreateUserWithEmailAndPassword“ čiji kod je prikazan na slici 4.2. Metodi se predaje e-mail adresa te lozinka s kojima se kreira korisnički račun u Firebase-u s jedinstvenim UserID-em. Firebase Auth sustav provjerava format unesene adrese e-pošte. Ukoliko je format adrese e-pošte ispravan, sustav provjerava unesenu lozinku koja mora sadržavati najmanje 6 znakova. Ukoliko u Firebase sustavu već postoji korisnik s navedenom adresom e-pošte, na zaslonu se prikazuje *Toast* poruka „User already exists.“

```
FirebaseAuth.getInstance().createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(
        OnCompleteListener<AuthResult>{task ->
            if(task.isSuccessful){
                val firebaseUser: FirebaseUser = task.result!!.user!!
                Toast.makeText(
                    context: this@RegisterActivity,
                    text: "You are registered successfully!",
                    Toast.LENGTH_SHORT
                ).show()

                var bmi_base = weight/(height.toDouble()*height.toDouble()/10000)
                val bmi:Double = String.format("%.2f", bmi_base).toDouble()
                CreateUser(firebaseUser.uid, email, password, name, age, weight, height, bmi, injury)

                val intent = Intent( packageContext: this@RegisterActivity, MainActivity::class.java)
                intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK or Intent.FLAG_ACTIVITY_CLEAR_TASK
                intent.putExtra( name: "user_id", firebaseUser.uid)
                intent.putExtra( name: "email_id", email)
                startActivity(intent)
                finish()
            }else{
                Toast.makeText(
                    context: this@RegisterActivity,
                    task.exception!!.message.toString(),
                    Toast.LENGTH_SHORT
                ).show()
            }
        })
    }
```

Slika 4.2 Metoda CreateUserWithEmailAndPassword

U navedenoj metodi „CreateUserWithEmailAndPassword“ poziva se metoda „CreateUser“ prikazana na slici 4.3 koja na temelju unesenih korisničkih parametara e-mail, lozinka, ime, dob, težina, visina i zdravstvenih ograničenja kreira korisnički račun. Među atributima metode „CreateUser“ nalazi se userID koji predstavlja jedinstveni ID trenutno registriranog korisnika. Među atributima nalazi se i BMI korisnika tj. indeks tjelesne mase koji se u metodi



„CreateUserWithEmailAndPassword“ računa na temelju unesene težine i visine. Iza userID nalaze se svi parametri tog korisnika.

```
private fun InsertUser(user: User,userID: String){
    val userReference=db.collection( collectionPath: "Users")

    userReference.document(userID).set(user).addOnCompleteListener{ it: Task<Void!>
        when{
            it.isSuccessful->{
                Toast.makeText( context: this, text: "User created",Toast.LENGTH_SHORT).show()
            }
            else ->{
                Toast.makeText( context: this, text: "User creation failed",Toast.LENGTH_SHORT).show()
            }
        }
    }
}

private fun CreateUser(userID:String,email:String,password:String, name:String,
    age:Int,weight:Double,height:Int, bmi: Double, injury:String){
    val user=User(userID,email, password, name, age, weight, height, bmi, injury)

    InsertUser(user,userID)
}
```

**Slika 4.3** Metode InsertUser i CreateUser

U metodi „CreateUser“ poziva se metoda „InsertUser“ koja pohranjuje trenutno registriranog korisnika u kolekciju „Users“ u Firebase Firestore. Ukoliko je pohrana korisnika uspješna, na zaslonu se prikazuje *Toast* poruka „User created“, u suprotnom se prikazuje *Toast* poruka „User creation failed“.

#### 4.2.2 Provjera potpunosti korisničkih parametara

Nakon što korisnik pri registraciji unese prijašnje navedene parametre predstavljene data klasom „User“ prikazane na slici 4.4, klikom na gumb „Registriraj se“ primjenom *Utility* funkcije „TextUtilis.IsEmpty“ dolazi do provjere potpunosti parametara prikazano na slici 4.5. Ukoliko korisnik nije ispunio polje za unos adrese e-pošte, na zaslonu se prikazuje *Toast* poruka „Molim Vas unesite email“. Navedeno vrijedi za unos svih parametara.

```
data class User(
    val userID:String,
    var email:String,
    var password:String,
    var name:String,
    var age:Int,
    var weight:Double,
    var height:Int,
    var bmi: Double,
    var injury: String
)
```

Slika 4.4 Data class User

```
btn_register.setOnClickListener { it:View!
    when {
        TextUtils.isEmpty(et_register_email.text.toString().trim { it <= ' ' }) -> {
            Toast.makeText(
                context: this@RegisterActivity,
                text: "Molim Vas unesite email!",
                Toast.LENGTH_SHORT
            ).show()
        }
        TextUtils.isEmpty(et_register_password.text.toString().trim { it <= ' ' })->{
            Toast.makeText(
                context: this@RegisterActivity,
                text: "Molim Vas unesite lozinku!",
                Toast.LENGTH_SHORT
            ).show()
        }
        TextUtils.isEmpty(et_register_name.text.toString().trim { it <= ' ' }) -> {
            Toast.makeText(
                context: this@RegisterActivity,
                text: "Molim Vas unesite ime!",
                Toast.LENGTH_SHORT
            ).show()
        }
        TextUtils.isEmpty(et_register_age.text.toString().trim { it <= ' ' })->{
            Toast.makeText(
                context: this@RegisterActivity,
                text: "Molim Vas unesite dob!",
                Toast.LENGTH_SHORT
            ).show()
        }
    }
}
```

Slika 4.5 Provjera popunjenosti korisničkih parametara

### 4.2.3 Spremanje korisnika u bazu podataka

Nakon što korisnik uspješno unese sve parametre i registrira se, njegovi podaci spremaju se u Firebase Firestore. Firestore se koristi u svrhe spremanja i kasnijeg dohvaćanja podataka o korisniku. Podaci o korisniku spremaju se u kolekciju „Users“. Svaki korisnik predstavljen je jedinstvenim ID-em koji se poklapa s ID-em u Firebase Auth. Prikaz podataka korisničkog računa u sustavu Firebase Auth prikazan je na slici 4.6.

## Authentication

Users   Sign-in method   Templates   Usage   Settings

Add user
↻
⋮

Identifier	Providers	Created ↓	Signed In	User UID
knezevic@gmail.com	✉	Aug 23, 2022	Aug 28, 2022	ViCbu01j5HW9RyVqa405pL0rZeS2
user@gmail.com	✉	Aug 22, 2022	Aug 22, 2022	JHWc9laPCFdxsQANfiV1ivn1acA2
user1@gmail.com	✉	Aug 21, 2022	Aug 26, 2022	4iu6xw5UlsRcdGB4k4sPioYREJu2
user55@gmail.com	✉	Aug 20, 2022	Aug 22, 2022	K2hfmoVy8e0geNtv6tZnS2JQYtu1

**Slika 4.6** Prikaz registriranih korisnika u Firebase Auth

U zbirci „Users“ nalaze se dokumenti koji predstavljaju jedinstvenog korisnika. Svaki dokument označen je korisnikovim jedinstvenim UserID-em. U svakom tom dokumentu nalaze se parametri tj. podaci tog korisnika. U dokumentu također se nalazi dodatna zbirka „History“ koja sadržava detalje o povijesti tjelovježbe korisnika. Prikaz baze podataka Firestore nalazi se na slici 4.7.

🏠 > Users > suxILsjdtkVLfQX.
More in Google Cloud

<div style="border-bottom: 1px solid #ccc; padding: 5px;"> <span>zavrnsni-43a0b</span> </div> <div style="padding: 5px;"> <span style="color: #007bff;">+ Start collection</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>Users &gt;</span> </div>	<div style="border-bottom: 1px solid #ccc; padding: 5px;"> <span>Users</span> </div> <div style="padding: 5px;"> <span style="color: #007bff;">+ Add document</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>4iu6xw5UlsRcdGB4k4sPioYREJu2</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>K2hfmoVy8e0geNtv6tZnS2JQYtu1</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>WiqY09ZAAPzOa5H0HEdBe2wbfIg1</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>bXjh0HxNehh0KH1YdorDLCZr0jj2</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>hLqY30BeNbZFM6BiS2j90rctQaz2</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span style="background-color: #e0e0e0;">suxILsjdtkVLfQX7GoZHszNHZR03 &gt;</span> </div>	<div style="border-bottom: 1px solid #ccc; padding: 5px;"> <span>suxILsjdtkVLfQX7GoZHszNHZR03</span> </div> <div style="padding: 5px;"> <span style="color: #007bff;">+ Start collection</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>History</span> </div> <div style="padding: 5px;"> <span style="color: #007bff;">+ Add field</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>age: 25</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>bmi: 22.49</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>email: "user99@gmail.com"</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>height: 170</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>injury: "Ozljeda koljena"</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>name: "Nina"</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>password: "123456"</span> </div> <div style="padding: 5px; border-bottom: 1px solid #ccc;"> <span>userID: "suxILsjdtkVLfQX7GoZHszNHZR03"</span> </div> <div style="padding: 5px;"> <span>weight: 65</span> </div>
--	---	--

**Slika 4.7** Prikaz korisničkih parametara u bazi podataka

#### 4.2.4 Prikaz korisničkih parametara

Na korisničkom profilu prikazani su određeni podaci korisnika kao što su ime, zdravstveno ograničenje, kilaža, visina i dob. Navedeni podaci dohvaćaju se iz baze podataka na temelju ID-a trenutno prijavljenog korisnika koji je predstavljen sa „currentUserReference“. Uz pomoć te

reference kreiramo novu referencu „currentUser“ te odabiremo putanju kolekcije i dokumenta iz kojeg želimo dohvatiti podatke. Dohvaćanje i prikaz korisničkih parametara nalazi se na slici 4.8.

```
val currentUserReference = FirebaseAuth.getInstance().uid.toString()
val currentUser = db.collection(collectionPath: "Users").document(currentUserReference)

currentUser.get().addOnSuccessListener { result ->
    currentUserName.text = result.data?.get("name").toString()
    currentUserInjury.text = result.data?.get("injury").toString()
    currentUserWeight.text = result.data?.get("weight").toString()
    currentUserHeight.text = result.data?.get("height").toString()
    currentUserAge.text = result.data?.get("age").toString()
}
```

Slika 4.8 Dohvaćanje korisničkih parametara iz baze podataka

#### 4.2.5 Obrada unesenih podataka

Koristeći prethodno navedenu referencu „currentUser“, obrađuje se korisnikov indeks tjelesne mase iz baze podataka. Indeks tjelesne mase obrađen je u funkciji „CreateUserWithEmailAndPassword“ na temelju korisnikove visine i težine. Metoda za filtriranje prikazana na slici 4.9 koristi ugrađenu funkciju *when* i na temelju korisnikovog BMI-a kreira pet mogućih ishoda. Ukoliko korisnik ažurira svoju visinu i težinu, ažurira se i njegov BMI te funkcija vraća drugi ishod ukoliko se parametar težine drastično promjeni. Na Home stranici aplikacije prikazuje se ishod filtriranja te korisniku daje informacije o njegovom indeksu tjelesne mase.

```
currentUser.get().addOnSuccessListener { result ->
    currentUserBMI.text = result.data?.get("bmi").toString()
    when(result.data?.get("bmi") as Double) {
        in 0.0..17.0 -> currentUserBMIstatus.text = "Vaš BMI je pre nizak"
        in 17.0..19.0 -> currentUserBMIstatus.text = "Vaš BMI je nizak"
        in 19.0..25.0 -> currentUserBMIstatus.text = "Vaš BMI je normalan"
        in 25.0..30.0 -> currentUserBMIstatus.text = "Vaš BMI je visok"
        else -> currentUserBMIstatus.text = "Vaš BMI je pre visok"
    }
}
```

Slika 4.9 Metoda za filtriranje indeksa tjelesne mase

#### 4.2.6 Prikaz preporuka

Pod istom referencom „currentUser“ filtriraju se preporučeni programi tjelovježbi. Metoda za filtriranje koristi ugrađenu funkciju *when* i na temelju unesene korisnikove ozljede kreira ukupno 18 mogućih ishoda, odnosno tri moguća ishoda za svaku ozljedu. Svakom korisniku se preporučuju 3 programa tjelovježbi ovisno o ozljedi. Na slici 4.10 prikazan je kod za filtriranje programa tjelovježbi ovisno o ozljedi. Iz koda vidimo ukoliko korisnik ima zdravstveno

ograničenje Dijabetes, preporučuju mu se program za dijabetes, program za istezanje i program za kardio. Navedeno vrijedi za ostala zdravstvena ograničenja. Svako zdravstveno ograničenje rezultira različitim preporukama programa tjelovježbe.

```
when(result.data?.get("injury")) {
  "Dijabetes" -> {
    tv_workout_title.text = "Program za dijabetes";
    tv_workout_title2.text = "Program za istezanje";
    tv_workout_title3.text = "Program za kardio";
  }
  "Invaliditet" -> {
    tv_workout_title.text = "Program za leđa i biceps";
    tv_workout_title2.text = "Program za ramena";
    tv_workout_title3.text = "Program za prsa i triceps";
  }
  "Ozljeda koljena" -> {
    tv_workout_title.text = "Program za leđa i biceps";
    tv_workout_title2.text = "Program za ramena";
    tv_workout_title3.text = "Program za prsa i triceps";
  }
  "Ozljeda leđa" -> {
    tv_workout_title.text = "Program za kvadriceps";
    tv_workout_title2.text = "Program za zadnju ložu";
    tv_workout_title3.text = "Program za kardio";
  }
  "Nema" -> {
    tv_workout_title.text = "Program za donji dio tijela";
    tv_workout_title2.text = "Program za gornji dio tijela";
    tv_workout_title3.text = "Program za kardio";
  }
  else -> {
    tv_workout_title.text = "Program za donji dio tijela";
    tv_workout_title2.text = "Program za gornji dio tijela";
    tv_workout_title3.text = "Program za kardio";
  }
}
```

**Slika 4.10** Kod za filtriranje programa tjelovježbi

Metoda „selectWorkoutFragment“ prikazana je na slici 4.11. Toj metodi predaje se Fragment koji želimo prikazati te vodi korisnika na navedeni Fragment. Metoda „selectWorkoutFragment“ zamjenjuje trenutni fragment *flFragment* s fragmentom *Fragment* koji predstavlja predani atribut toj metodi. Koristeći tu metodu na temelju korisnikove ozljede, klik na gumb koji se nalazi ispod preporučenog programa tjelovježbe vodi korisnika na preporučeni *Fragment*. Na slici 4.12 prikazano je filtriranje Fragmenta na temelju korisnikove ozljede.

```
private fun selectWorkoutFragment(fragment: Fragment){
    val transaction = requireActivity().supportFragmentManager.beginTransaction()
    transaction.replace(R.id.flFragment, fragment)
    transaction.disallowAddToBackStack()
    transaction.commit()
}
```

**Slika 4.11** Metoda za izmjenu Fragmenta

```
when(result.data?.get("injury")) {
    "Invaliditet" -> {
        btn_start_workout.setOnClickListener() { selectWorkoutFragment(BackWorkoutFragment()) }
        btn_start_workout2.setOnClickListener() { selectWorkoutFragment(ShoulderWorkoutFragment()) }
        btn_start_workout3.setOnClickListener() { selectWorkoutFragment(ChestWorkoutFragment()) }
    }
}
```

**Slika 4.12** Filtriranje Fragmenta na temelju korisnikove ozlijede

#### 4.2.7 Spremanje povijesti vježbanja u bazu podataka

Na detaljnom prikazu programa tjelovježbe nalazi se gumb kojim korisnik označava tjelovježbu kao odrađenu. Na slici 4.13 prikazana je ugrađena metoda „setOnClickListener“ koja klikom na gumb sprema podatke proslijeđene kao parametre metodi „saveFirestore“ i prikazuje *Toast* poruku „Added to history“.

```
btn_add_history.setOnClickListener(){ it: View!
    val workoutTitle = tv_workout_title.text.toString()
    val workoutCalories = tv_workout_calories.text.toString()
    saveFirestore(workoutTitle, workoutCalories)
    Toast.makeText(context, text: "Added to history", Toast.LENGTH_SHORT).show()
}
```

**Slika 4.13** Kod za spremanje programa vježbe u bazu podataka

Metoda „saveFirestore“ prikazana na slici 4.14 dohvaća referencu trenutnog korisnika te u bazu podataka sprema proslijeđene parametre ime tjelovježbe i broj potrošenih kalorija. Uz te podatke također i sprema datum odrade tjelovježbe u formatu „DD/MM/YYYY“

```

fun saveFirestore(workoutTitle: String, workoutCalories: String){
    val currentUserReference = FirebaseAuth.getInstance().uid.toString()
    val db = FirebaseFirestore.getInstance()

    val df = SimpleDateFormat( pattern: "dd/MM/yyyy")
    val cal = Calendar.getInstance()
    val dateFormat = df.format(cal.time)

    val workout: MutableMap<String, Any> = HashMap()
    workout["workoutTitle"] = workoutTitle
    workout["date"] = dateFormat
    workout["calories"] = workoutCalories

    db.collection( collectionPath: "Users").document(currentUserReference).collection( collectionPath: "History").add(workout)
}

```

Slika 4.14 Metoda za spremanje programa tjelovježbe u bazu podataka

#### 4.2.8 Prikaz povijesti vježbanja

Na korisničkom profilu prikazana je povijest odrađenih programa tjelovježbi. Na temelju prethodno spremljenih podataka u kolekciju „History“ pod dokumentom „currentUserReference“ koja predstavlja UserID trenutno prijavljenog korisnika, prikazuju se podaci o imenu programa tjelovježbe, datuma odrade i broju potrošenih kalorija. Prikaz podataka filtrira se prema datumu odrade od najnovijih prema najstarijim. Filtriranje je omogućeno ugrađenom metodom „orderBy“ kojoj predajemo parametar po kojem želimo filtrirati kolekciju, u ovom slučaju „date“. Na slici 4.15 nalazi se kod za prikaz povijesti vježbanja.

```

val db = FirebaseFirestore.getInstance()
db.collection( collectionPath: "Users").document(currentUserReference).collection( collectionPath: "History")
    .orderBy( field: "date", Query.Direction.DESCEENDING)
    .get()
    .addOnCompleteListener { it: Task<QuerySnapshot!>

        val title: StringBuffer = StringBuffer()
        val date: StringBuffer = StringBuffer()
        val calories: StringBuffer = StringBuffer()

        if(it.isSuccessful) {
            for(document in it.result!!) {

                title.append(document.data.getValue( key: "workoutTitle" )).append("\n\n")
                date.append(document.data.getValue( key: "date" ).toString()).append("\n\n")
                calories.append(document.data.getValue( key: "calories" ).toString()).append("\n\n")
            }
            workoutHistory.setText(title)
            workoutDate.setText(date)
            workoutCalories.setText(calories)
        }
    }
}

```

Slika 4.15 Kod za prikaz povijesti tjelovježbe

## 4.2.9 Ažuriranje korisničkih parametara

Na korisničkom profilu nalazi se gumb „Uredi“ koji korisniku omogućuje ažuriranje unesenih podataka, točnije ažuriranje težine i visine. Prvo se provjerava jesu li oba polja unesena, ako nisu na zaslonu se prikazuje *Toast* poruka „Molim Vas unesite težinu/visinu“. Nakon što korisnik unese ažurirane podatke, ažurira se i indeks tjelesne mase na temelju unesenih podataka. Ažurirani podaci se spremaju u bazu podataka ugrađenom metodom „update“, tj. zamjenjuje prethodno unesene podatke. Na slici 4.16 prikazano je ažuriranje korisničkih podataka.

```
btn_edit_db.setOnClickListener() { it: View!
    when {
        TextUtils.isEmpty(et_edit_weight.text.toString().trim { it <= ' ' }) -> {
            Toast.makeText(activity, text: "Molim Vas unesite težinu!", Toast.LENGTH_SHORT)
                .show()
        }
        TextUtils.isEmpty(et_edit_height.text.toString().trim { it <= ' ' }) -> {
            Toast.makeText(activity, text: "Molim Vas unesite visinu!", Toast.LENGTH_SHORT)
                .show()
        }
    }

    else -> {
        var updatedWeight = et_edit_weight.text.toString().toDouble()
        db.collection( collectionPath: "Users").document(currentUserReference).update( field: "weight", updatedWeight)

        var updatedHeight = et_edit_height.text.toString().toDouble()
        db.collection( collectionPath: "Users").document(currentUserReference).update( field: "height", updatedHeight)

        var updatedBMI_base = updatedWeight / (updatedHeight.toDouble() * updatedHeight.toDouble() / 10000)
        val updatedBMI: Double = String.format("%.2f", updatedBMI_base).toDouble()
        db.collection( collectionPath: "Users").document(currentUserReference).update( field: "bmi", updatedBMI)
    }
}
```

Slika 4.16 Kod za ažuriranje korisničkih parametara



## 5. NAČIN KORIŠTENJA I ISPITIVANJE MOBILNE APLIKACIJE S ANALIZOM REZULTATA

U ovom poglavlju opisuje se način korištenja mobilne aplikacije, postupci kroz koje korisnik prolazi te prikaz komponenti aplikacije.

### 5.1 Korištenje mobilne aplikacije

Prvi korak korištenja mobilne aplikacije je instaliranje. Nakon instaliranja prikazan je zaslon za prijavu korisnika u sustav gdje korisnik unosi adresu e-pošte i lozinku. Ukoliko korisnik nema korisnički račun, klikom na gumb „Registriraj se“ otvara se zaslon za registriranje. Na zaslonu za registriranje od korisnika se traži unos adrese e-pošte u ispravnom formatu te unos lozinke koja sadrži najmanje šest znakova. Na ovom zaslonu korisnik unosi sljedeće parametre: ime, dob, težina, visina i zdravstveno ograničenje. Ako korisnik nije unio barem jedan od navedenih parametara, na zaslonu se prikazuje *Toast* poruka koja ga upozorava kako nije unio sve parametre. Kada korisnik unese sve parametre i uspješno se registrira, otvara se glavni prozor aplikacije koji ima navigaciju na vrhu koja sadrži tri dijela: korisnički profil, početni zaslon te odjava korisnika iz sustava.

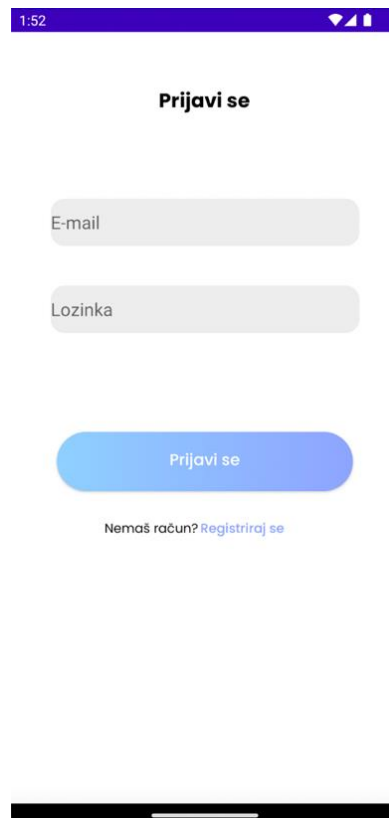
Na početnom zaslonu nalazi se korisnikov indeks tjelesne mase koji je izračunat na temelju unesene težine i visine. Korisnik također ima uvid u stanje njegovog indeksa tjelesne mase koje može biti nisko, normalno ili visoko. Ispod uvida na indeks tjelesne mase nalaze se preporučeni programi tjelovježbe za tog korisnika. Preporučeni programi tjelovježbe filtrirani su na temelju unesenog zdravstvenog ograničenja pri registraciji. Tu je prikazan naziv programa, broj vježbi trajanje i gumb „Započni“. Klikom na gumb otvara se zaslon programa tjelovježbe na kojem je prikazan popis vježbi. Svaka vježba opisana je nazivom, brojem ponavljanja, ilustracijom koja upućuje korisnika na pravilno izvođenje te linkom kojim se otvara YouTube video tutorijal. Na vrhu zaslona nalazi se gumb „Označi kao odrađeno“. Pritiskom na taj gumb informacije tog programa spremaju se u bazu podataka i prikazuju se na korisničkom profilu.

Na zaslonu korisničkog profila nalaze se informacije o trenutno prijavljenom korisniku kao što su ime, zdravstveno ograničenje, težina, visina i dob. Ispod navedenih informacija nalazi se korisnikova povijest vježbanja s datumom odrade programa te brojem potrošenih kalorija. Na zaslonu korisničkog profila nalazi se gumb „Uredi“ kojim korisnik ima mogućnost ažurirati svoju težinu i visinu. Kada korisnik ažurira svoju težinu i visinu, automatski se ažurira i njegov indeks tjelesne mase. U navigaciji se nalazi gumb za odjavu kojim se korisnik odjavljuje iz sustava i otvara se zaslon za prijavu.

## 5.2 Prikaz rada mobilne aplikacije

### 5.2.1 Prikaz prijave korisnika

Pri prvom pokretanju mobilne aplikacije prikazuje se zaslon za prijavu korisnika prikazan na slici 5.1. Od korisnika traži unos adresu e-pošte u lozinku. Za uspješnu prijavu potrebna je adresa e-pošte u ispravnom formatu i lozinka koja sadrži minimalno šest znakova. Klikom na gumb „Prijavi se“ šalje se zahtjev za prijavu u bazu. Ako u bazi podataka Firebase Auth postoji račun s unesenim podacima, korisnik je uspješno prijavljen i na zaslonu mu se prikazuje početna stranica. Ukoliko u bazi podataka ne postoji račun s unesenim podacima, prikazuje se *Toast* poruka da je prijava neuspješna jer ne postoji račun s tim podacima. U slučaju da korisnik nema prijavljen račun, klikom na gumb „Registriraj se“ prosljeđen je na zaslon za registraciju.



Slika 5.1 Sučelje prijave

### 5.2.2 Prikaz registracije i unos korisničkih parametara

Ukoliko korisnik nema prijavljen račun, prosljeđuje se na zaslon za registraciju prikazan na slici 5.2. Prvo se traži unos adrese e-pošte u ispravnom formatu te lozinka koja sadrži najmanje šest znakova. Također se traži unos sljedećih podataka: ime, dob, kilaža, visina te zdravstvena ograničenja. Ukoliko korisnik nije popunio sve podatke, na zaslonu se prikazuje *Toast* poruka koja ga upozorava da nije ispunio sva polja.

1:53 3G

### Registriraj se

Imas račun? [Prijavi se](#)

E-mail

Lozinka

Ime

Dob

Kilaža (u kg)

Visina (u cm)

Zdravstvena ograničenja ▾

Registriraj se

**Slika 5.2** Sučelje registracije

### 5.2.3 Prikaz početnog zaslona

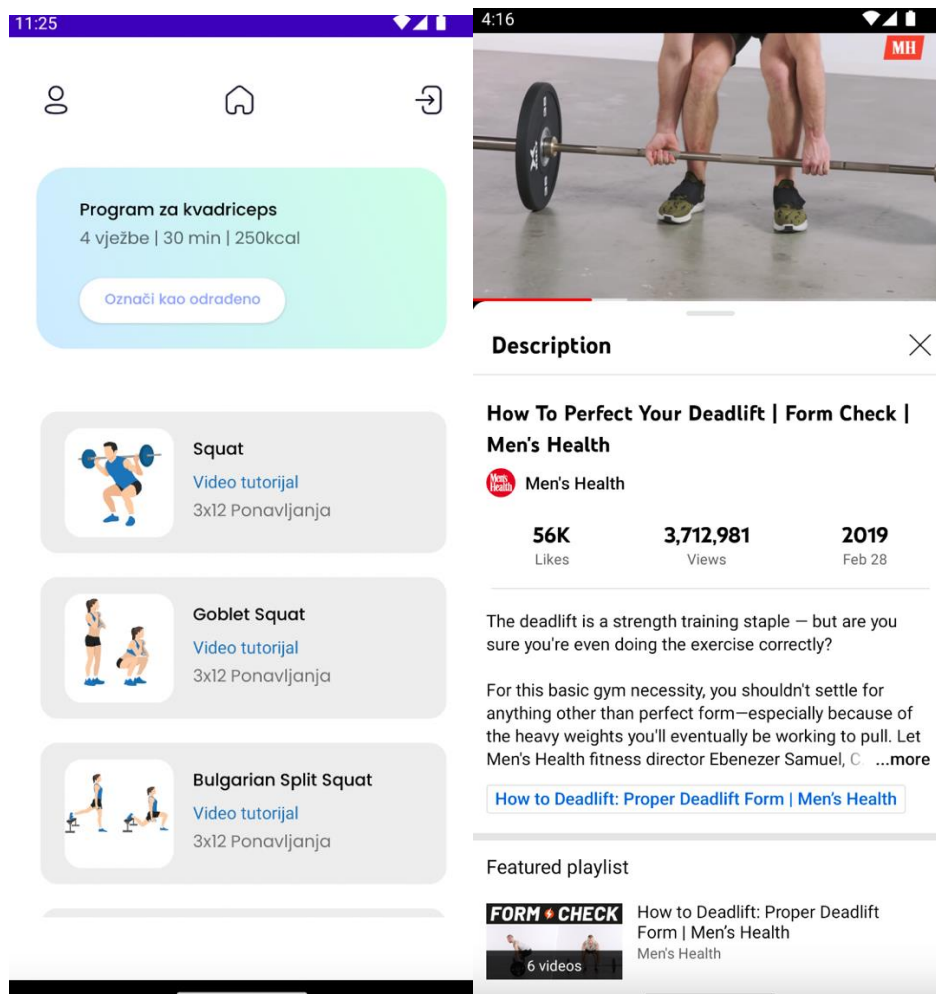
Nakon što se korisnik prijavi odnosno registrira, dolazi do početnog zaslona prikazanog na slici 5.3. Na vrhu zaslona nalazi se navigacija koja sadrži tri gumba. Prvi gumb predstavljen ikonom *Profile* vodi korisnika na svoj korisnički profil. Drugi gumb predstavljen ikonom *Home* vodi korisnika na početni zaslon. Zadnji gumb predstavljen ikonom *Logout* odjavljuje korisnika iz sustava te ga vodi na zaslon za prijavu. Ispod navigacije nalazi se kartica koja sadrži podatke o indeksu tjelesne mase korisnika. Na temelju trenutnog indeksa tjelesne mase formiran je *String* koji korisniku daje informaciju dali je njegov indeks tjelesne mase nizak, normalan ili visok. Ispod kartice s indeksom tjelesne mase nalaze se preporučene vježbe za korisnika na temelju unesenih podataka pri prijavi. Korisniku su preporučena tri programa tjelovježbe od kojih je svaki predstavljen odvojenom karticom. Na svakoj kartici programa tjelovježbe nalaze se najbitniji podaci o tom programu kao što su ime programa, broj tjelovježbi, trajanje vježbi i prosječan broj potrošenih kalorija tjelovježbe. Klikom na gumb „Započni“, korisnik se prebacuje na zaslon programa tjelovježbe.



Slika 5.3 Sučelje početnog zaslona

#### 5.2.4 Prikaz programa tjelovježbe

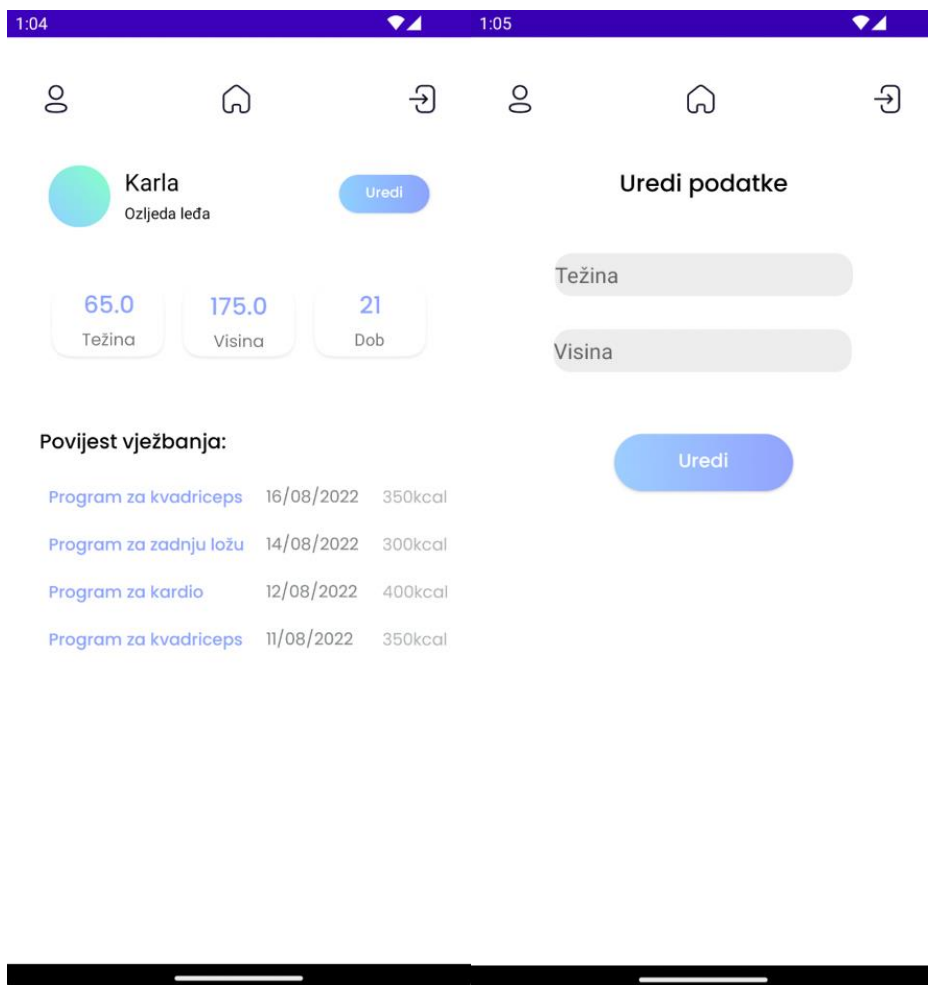
Nakon što se korisnik odabere program tjelovježbe prebacuje se na zaslon prikazan na slici 5.4. Na vrhu zaslona programa tjelovježbe nalaze se najbitnije informacije o programu kao što su ime programa, broj vježbi, trajanje programa i broj potrošenih kalorija. Tu se također nalazi i gumb „Označi kao odrađeno“. Klikom na gumb spremaju se podaci o programu tjelovježbe u bazu podataka u kolekciju „History“. Ukoliko je spremanje u bazu podataka uspješno, na zaslonu se pojavljuje *Toast* poruka „Added to History“. Ispod se nalazi popis tjelovježbi. Svaka vježba predstavljena je karticom na kojoj se nalazi ime vježbe i ilustracija koja korisnika upućuje na izgled vježbe. Ukoliko korisnik nije siguran kako pravilno izvesti navedenu vježbu, ispod imena vježbe nalazi se link „Video tutorijal“ koji otvara YouTube video na kojem je prikazano pravilno izvođenje vježbe. Ispod linka za YouTube video, nalazi se broj ponavljanja vježbe. Za svaku vježbu idealno je preporučeno 3 skupa po 12 ponavljanja.



**Slika 5.4** Sučelje programa tjelovježbe

### 5.2.5 Prikaz korisničkog profila

Prva ikona na navigaciji početnog zaslona otvara zaslon korisničkog profila. Tu su prikazane sljedeće informacije o korisniku: ime, zdravstveno ograničenje, težina, visina i dob. Pored korisnikovog imena nalazi se gumb „Uredi“. Pritiskom se otvara novi zaslon na kojem korisnik ima mogućnost ažurirati svoje parametre težine i visine. Kada korisnik unese podatke, klikom na gumb „Uredi“, njegovi podaci se ažuriraju u bazi podataka i na profilu. Ukoliko ih korisnik ne želi ažurirati, pritiskom na ikonu *Profile* vraća se na korisnički profil. Ispod informacija o trenutno prijavljenom korisniku nalazi se povijest vježbanja. Sastoji se od popisa kojeg čine tri elementa: naziv programa tjelovježbe, datum odrade tjelovježbe i broj potrošenih kalorija. Uvidom u učestalost vježbanja motivira korisnika da svakodnevno vodi aktivan život. Na slici 5.5. prikazano je korisničko sučelje profila.



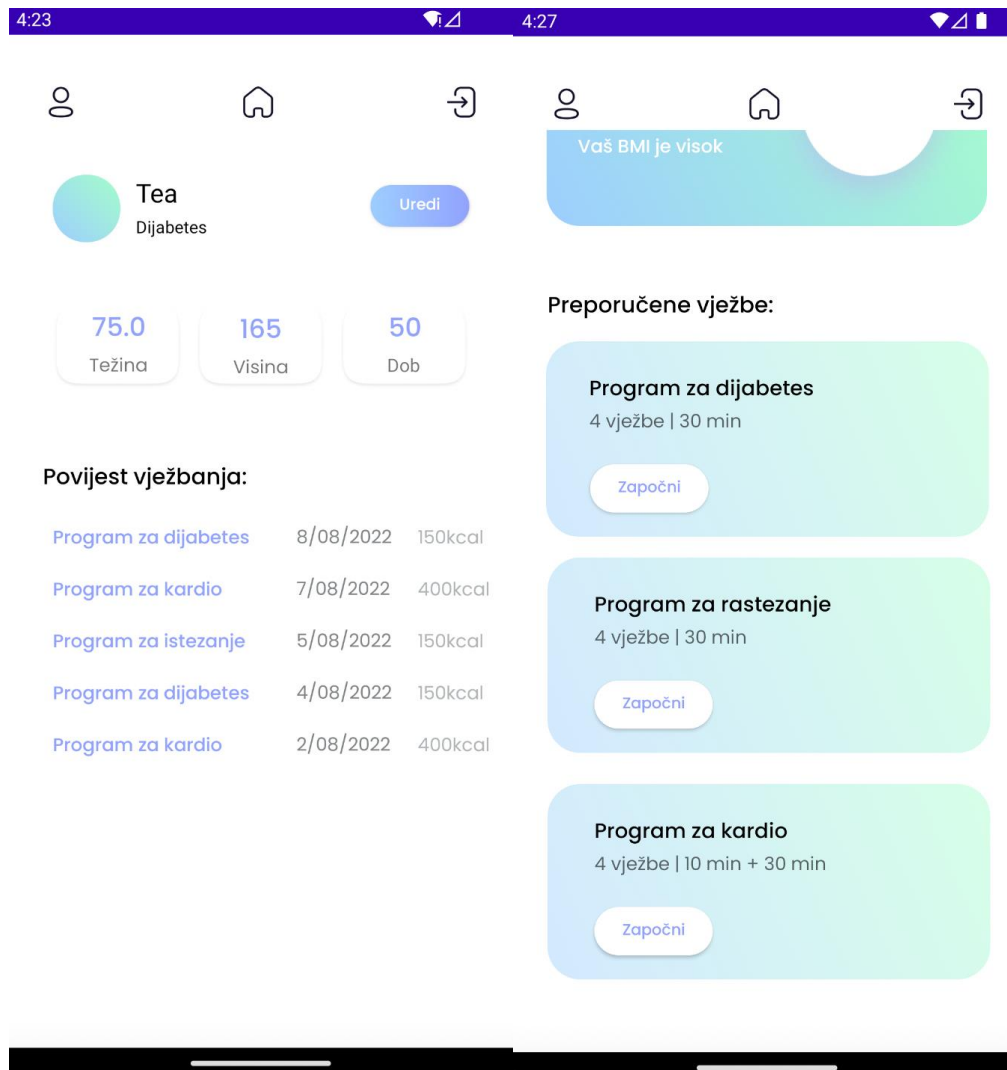
Slika 5.5 Sučelje korisničkog profila

### 5.3. Ispitivanje rada mobilne aplikacije

Opisana su tri slučaja ispitivanja rada aplikacije. Riječ je o slučaju za korisnika s dijabetesom, korisnika s ozljedom koljena i korisnika bez zdravstvenih ograničenja. Svaki od navedenih slučajeva ima posebne karakteristike i rezultira različitim preporukama. Svaki slučaj prikazan je korisničkim sučeljem profila i početnog zaslona s preporučenim programima tjelovježbe.

#### 5.3.1. Korisnički slučaj – Dijabetes

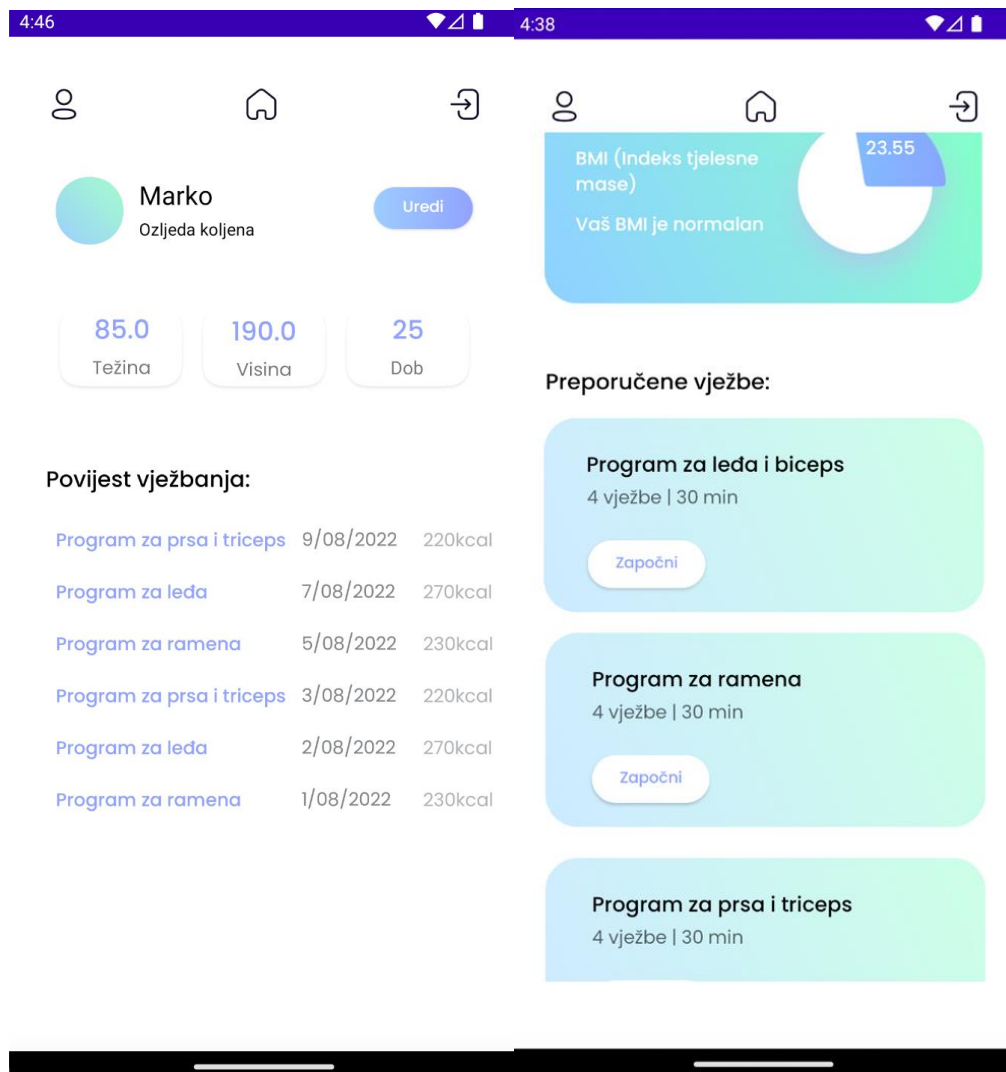
Za razliku od drugih kroničnih bolesti, za osobe oboljele od dijabetesa dopušteno je sudjelovanje u gotovo svim oblicima tjelovježbe uz razinu opreznosti. Na primjer, ako razina šećera u krvi nije optimalna, moguće su posljedice nakon vježbanja visokog intenziteta. Uz to je bitno nositi prikladnu odjeću i obuću za udobnost. Stručnjaci tvrde da bi učinkovit program vježbanja za osobe s dijabetesom trebao uključivati kardio vježbe, vježbe snage i fleksibilnosti. Korisniku oboljelog od dijabetesa preporučuju se sljedeći programi tjelovježbe: program za dijabetes, program za rastezanje i program za kardio. Na slici 5.6 prikazano je korisničko sučelje za slučaj zdravstvenog ograničenja dijabetes.



**Slika 5.6** Korisničko sučelje za slučaj zdravstvenog ograničenja dijabetes

### 5.3.2. Korisnički slučaj – Ozljede koljena

Bolovi i ozljede koljena mogu imati mnogo uzroka. To mogu biti jednostavni pokreti poput običnog hodanja ili trčanja ili složeniji pokreti poput skijanja i natjecateljskih sportova. Ozljede nastaju zbog slabosti i neaktivnosti mišića ekstenzora, fleksora i stabilizatora koljenskog zgloba, pa se sva sila i težina tijekom pokreta prenosi na tetive i kosti koljenskog zgloba. Osobama s ozljedom koljena propisana je liječnička terapija za oporavak koljena. Ukoliko ta osoba želi nastaviti treniranjem ostatka tijela, preporučaju se vježbe za gornji dio tijela u sjedećem položaju tj. bez opterećenja donjem dijelu tijela i koljena. Ovom korisniku preporučaju se sljedeći programi tjelovježbe: program za leđa i biceps, program za ramena te program za prsa i triceps. Na slici 5.7 prikazano je korisničko sučelje za slučaj zdravstvenog ograničenja ozljeda koljena.

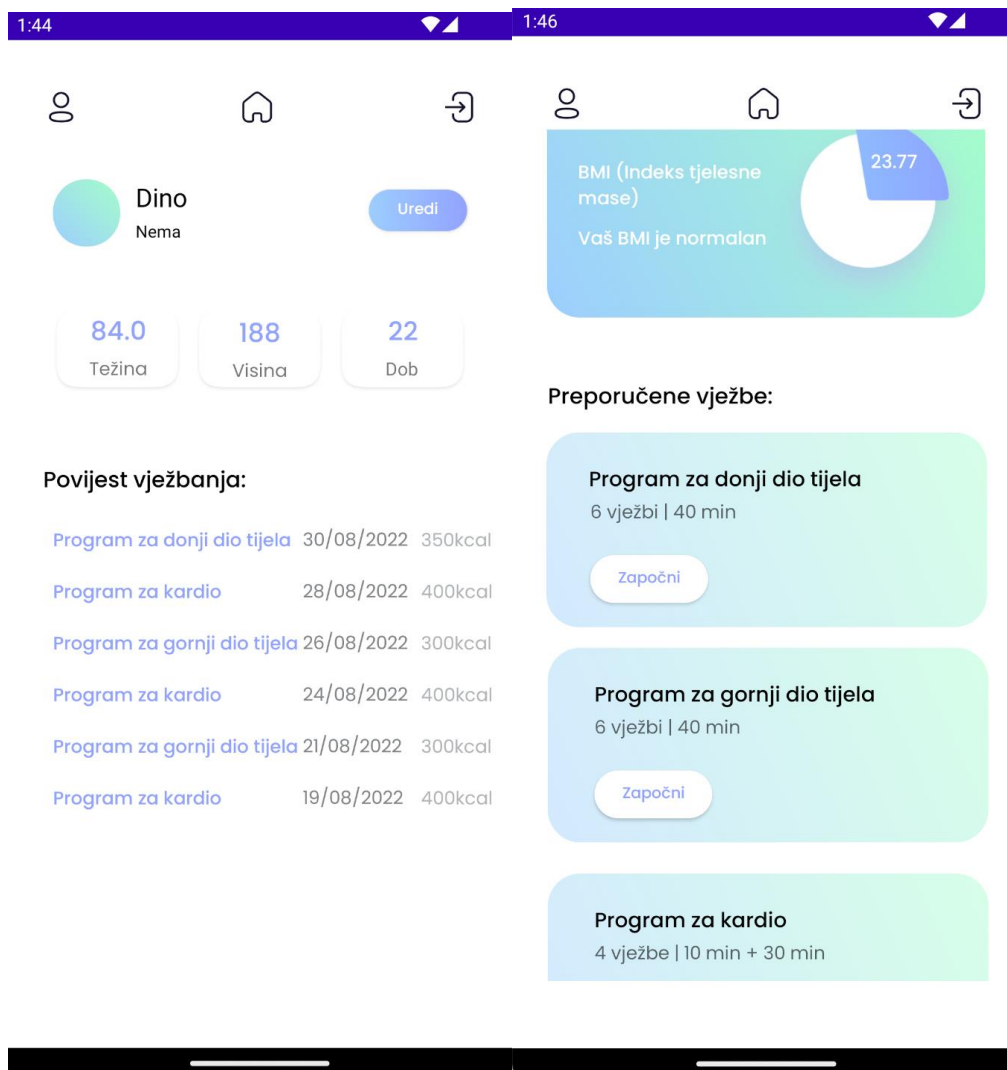


Slika 5.7 Korisničko sučelje za slučaj zdravstvenog ograničenja ozljeda koljena

### 5.3.3 Korisnički slučaj – Nema zdravstvenih ograničenja

Najidealniji slučaj je za korisnika koji nema zdravstvenih ograničenja. Taj korisnik može izvoditi sve vježbe za svaki dio tijela. Nema rizika za potencijalnu ozljedu ukoliko sve vježbe u programu izvodi pravilnom formom. Pravilno izvođenje svake vježbe prikazano je ilustracijom u detaljima programa tjelovježbe, te YouTube video linkom. Ovom korisniku preporučeni su programi za svaki dio tijela. To su program za donji dio tijela, program za gornji dio tijela i kardio program. Na slici 5.8 prikazano je korisničko sučelje za slučaj korisnika bez zdravstvenog ograničenja.





Slika 5 8 Korisničko sučelje za korisnika bez zdravstvenih ograničenja

#### 5.4 Analiza rezultata ispitivanja mob aplikacije

Svaki od ispitanih korisničkih slučajeva imao je različiti ishod. Osobi oboljeloj od dijabetesa preporučeni su manje intenzivni programi tjelovježbe kako bi izbjegli negativne posljedice koje nastaju zbog velikog intenziteta. Korisniku s ozljedom koljena preporučeni su programi fokusirani na gornji dio tijela kako bi se donji dio tijela tj. ozljeda koljena uspješno oporavila. Programi preporučeni tom korisniku izvode se uglavnom u sjedećem položaju kako bi zaobišli još veću potencijalnu ozljedu na donjem dijelu tijela. Najidealniji od ispitanih korisničkih slučajeva je zadnji, korisnik bez zdravstvenih ograničenja. Taj korisnik ima slobodu izvođenja tjelovježbi željenim intenzitetom za svaki dio tijela te nema potencijalnih ozljeda.

## 6. ZAKLJUČAK

Cilj ovog rada je razvoj mobilne aplikacije za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti. Ova mobilna aplikacija razvijena je sa ciljem pronalaženja prikladnih programa tjelovježbi za svakog korisnika s obzirom na zdravstveno stanje. Također potiče korisnika na održavanje aktivnog i zdravog života. Jednostavno korisničko sučelje omogućuje registraciju i prijavu korisnika u sustav, višekriterijski sustav preporuka programa tjelovježbe te praćenja učestalosti vježbanja i aktivnosti korisnika. Mobilna aplikacija prikladna je za korisnike svih uzrasta i zdravstvenih ograničenja.

Mobilna aplikacija ostvarena je u razvojnoj okolini Android Studio koristeći programski jezik Kotlin, aplikaciji Figma za stvaranje korisničkog sučelja i platformi Firebase za stvaranje baze podataka. Ispitana je funkcionalnost aplikacije za različite korisnike i rezultira ispravnosti rada mobilne aplikacije. Vidljivo je da za različite unose aplikacija nudi prikladne programe tjelovježbe. Mobilna aplikacija može u budućnosti biti poboljšanja proširenjem baze s zdravstvenim ograničenjima i programima tjelovježbe.

## LITERARURA

- [1] Zdravlje srca i tjelesna aktivnost, <https://www.hzjz.hr/sluzba-epidemiologija-prevencija-nezaraznih-bolesti/zdravlje-srca-i-tjelesna-aktivnost/>, pristupljeno 20.06.2022.
- [2] Tjelesna aktivnost utječe na zdravlje, <https://nzjz-split.hr/tjelesna-aktivnost-utjece-na-zdravlje/>, pristupljeno 20.06.2022.
- [3] Zašto je važna tjelesna aktivnost, <https://skolica-prehrane.rijeka.hr/tjelesna-aktivnost/zasto-je-vazna-tjelesna-aktivnost/>, pristupljeno 20.06.2022.
- [4] Korisnost redovite tjelesne aktivnosti, <https://www.zzjzdnz.hr/zdravlje/tjelesna-aktivnost/333>, pristupljeno 20.06.2022.
- [5] M. Duraković, M. Tjelesno vježbanje i zdravlje, Znanje, 2018.
- [6] Tjelesna aktivnost u službi zdravlja, <http://www.stampar.hr/hr/tjelesna-aktivnost-u-sluzbizdravlja>, pristupljeno 20.06.2022.
- [7] S. R. Chipkin, S. A. Klugh, L. Chasan-Taber, Exercise and Diabetes, Cardiology Clinics, Volume 19, Issue 3, University of Massachusetts Amherst, 2001, pp. 489-505.
- [8] C. Bouchard, J-P.Depres, A. Tremblay, Exercise and Obesity, Obesity Research, 1993, pp. 133-147.
- [9] Invaliditet, <https://www.mentalnitrening.hr/2019/03/17/sport-osoba-s-invaliditetom-ili-zasto-bi-se-svi-trebali-baviti-sportom/>, pristupljeno 20.06.2022.
- [10] Home Workout, <https://play.google.com/store/apps/details?id=com.home.workouts.professional>, pristupljeno 20.06.2022.
- [11] Senior Fitness, <https://play.google.com/store/apps/details?id=fitness.com.senior>, pristupljeno 20.06.2022.
- [12] Injurymap, <https://play.google.com/store/apps/details?id=com.injurymap.injurymap>, pristupljeno 20.06.2022.
- [13] Q. Shambour, J. Lu, Integrating Multi-Criteria Collaborative Filtering and Trust filtering for personalized Recommender Systems, 2011. IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MDCM), France, 2011, pp. 44-5.

- [14] H. Mohamed, L. Abdulsalam, H. Mohammed, "Adaptive Genetic Algorithm for Improving Prediction Accuracy of a Multi-Criteria Recommender System," 2018 IEEE 12th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoc), Vietnam, 2018, pp. 79-86.
- [15] S. M. Al-Ghuribi, S. A. Mohd Noah, "Multi-Criteria Review-Based Recommender System–The State of the Art," in IEEE Access, vol. 7, 2019, pp. 169446-169468.
- [16] D. Rowley, The Business of Application Portability, University of Bamberg Press Bamberg, 1996, pp. 80-87.
- [17] A. Sarkar, A. Goyal, D. Hicks, D. Sarkar, S. Hazra, "Android Application Development: A Brief Overview of Android Platforms and Evolution of Security Systems," 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), India, 2019, pp. 73-79.
- [18] Android Architecture, <https://developer.android.com/guide/platform>, pristupljeno 20.06.2022.
- [19] Android Studio, <https://developer.android.com/studio/intro>, pristupljeno 20.06.2022.
- [20] Kotlin, <https://kotlinlang.org/>, pristupljeno 20.06.2022.
- [21] Figma, <https://www.figma.com/ui-design-tool/>, pristupljeno 20.06.2022.
- [22] Firebase, <https://firebase.google.com/>, pristupljeno 20.06.2022.

## POPIS SLIKA

<b>Slika 2.1</b> Sučelje aplikacije Home Workout .....	6
<b>Slika 2.2</b> Sučelje aplikacije Senior Fitness .....	7
<b>Slika 2.3</b> Sučelje aplikacije Injurymap .....	7
<b>Slika 3.1</b> Dijagram mobilne aplikacije za preporuku fizičke aktivnosti.....	14
<b>Slika 4.1</b> Korisničko sučelje alata Figma.....	16
<b>Slika 4.2</b> Metoda CreateUserWithEmailAndPassword .....	17
<b>Slika 4.3</b> Metode InsertUser i CreateUser .....	18
<b>Slika 4.4</b> Data class User .....	19
<b>Slika 4.5</b> Provjera potpunosti korisničkih parametara .....	19
<b>Slika 4.6</b> Prikaz registriranih korisnika u Firebase Auth .....	20
<b>Slika 4.7</b> Prikaz korisničkih parametara u bazi podataka .....	20
<b>Slika 4.8</b> Dohvaćanje korisničkih parametara iz baze podataka.....	21
<b>Slika 4.9</b> Metoda za filtriranje indeksa tjelesne mase.....	21
<b>Slika 4.10</b> Kod za filtriranje programa tjelesne mase .....	22
<b>Slika 4.11</b> Metoda za izmjenu Fragmenta .....	23
<b>Slika 4.12</b> Filtriranje Fragmenta na temelju korisnikove ozlijede .....	23
<b>Slika 4.13</b> Kod za spremanje programa vježbe u bazu podataka.....	23
<b>Slika 4.14</b> Metoda za spremanje programa tjelesne mase u bazu podataka .....	24
<b>Slika 4.15</b> Kod za prikaz povijesti tjelesne mase .....	24
<b>Slika 4.16</b> Kod za ažuriranje korisničkih parametara .....	25
<b>Slika 5.1</b> Sučelje prijave.....	27
<b>Slika 5.2</b> Sučelje registracije.....	28
<b>Slika 5.3</b> Sučelje početnog zaslona.....	29
<b>Slika 5.4</b> Sučelje programa tjelesne mase.....	30
<b>Slika 5.5</b> Sučelje korisničkog profila .....	31

<b>Slika 5.6</b> Korisničko sučelje za slučaj zdravstvenog ograničenja dijabetes .....	32
<b>Slika 5.7</b> Korisničko sučelje za slučaj zdravstvenog ograničenja ozljeda koljena .....	33
<b>Slika 5.8</b> Korisničko sučelje za korisnika bez zdravstvenih ograničenja .....	34

## **POPIS TABLICA**

<b>Tablica 3.1</b> Prikaz korisničkih parametara .....	10
<b>Tablica 3.2</b> Opis i oznaka programa vježbe .....	11
<b>Tablica 3.3</b> Preporuka programa na temelju ozljede .....	11

## SAŽETAK

U ovom završnom radu cilj je predložiti i ostvariti mobilnu aplikaciju za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti. Mobilna aplikacija ostvarena je u razvojnoj okolini Android Studio koristeći programski jezik Kotlin, aplikaciji Figma za stvaranje korisničkog sučelja i platformi Firebase za stvaranje baze podataka. Omogućuje registriranje, odnosno prijavu korisnika i unos korisničkih parametara i zdravstvenih ograničenja. Na temelju unesenih parametara, korisniku se predlažu preporučeni programi tjelovježbe idealne za njegovo zdravstveno ograničenje. Također, korisnik ima uvid u svoj indeks tjelesne mase te mogućnost izmjene svoje težine čime se mijenja i indeks tjelesne mase. Mobilna aplikacija za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti omogućuje pregled povijesti odrađenih vježbi te motivira korisnika da održi zdrav i aktivan život.

**Ključne riječi:** Android, fizičke aktivnosti, mobilna aplikacija, praćenje fizičkih aktivnosti, sustav stvaranja personaliziranih preporuka.



## **ABSTRACT**

Mobile Android application for multi-criteria personalized recommendation of physical activities.

In this final work, the goal was to propose and evolve a mobile application for multi-criteria personalized recommendation of physical activities. The mobile application was created in the Android Studio development environment using the Kotlin programming language, Figma application for creating a user interface, and Firebase platform for creating a database. It enables registration, i.e. user login and entry of user parameters and health restrictions. Based on the entered parameters, the user is suggested recommended exercise programs ideal for his health restriction. He also has insight into his body mass index, and has the option of changing his weight, which also automatically adjusts the body mass index. The mobile application for multi-criteria personalized recommendation of physical activities gives insight in the history of performed exercises and motivates the user to maintain a healthy and active life.

**Keywords:** Android, physical activities, mobile application, monitoring of physical activities, system for creating personalized recommendations.

## **ŽIVOTOPIS**

Karla Fehir rođena je 21. prosinca 2000. godine u Đakovu. Pohađala je Osnovnu školu Blage Zadre u Vukovaru. Upisuje Gimnaziju Vukovar 2015. godine te ju završava 2019. godine. Iste godine upisuje FERIT Osijek, preddiplomski studij računarstva.

## **PRILOZI**

Prilog 1. Završni rad „Mobilna aplikacija za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti“ u *.docx* formatu.

Prilog 2. Završni rad „Mobilna aplikacija za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti“ u *.pdf* formatu.

Prilog 3. Programsko ostvarenje mobilne aplikacije za višekriterijsko personalizirano preporučivanje fizičkih aktivnosti.