

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Klementina Garbajs

Mobilna aplikacija Asana Master

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mira Trebar

Ljubljana, 2021

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogu:

Tematika naloge:

V diplomskem delu zasnujte mobilno aplikacijo za osebno vadbo joge. Omogoča naj ogled in izvajanje položajev, spremeljanje napredka zastavljenih ciljev z beleženjem posameznih aktivnosti in občutkov ter shranjevanjem zapiskov in posnetkov. Predlagano rešitev implementirajte v ogrodju Expo z React Native in v okviru izbranega mobilnega operacijskega sistema Android predstavite njeno uporabo ter podajte nadaljnje možnosti razvoja.

Na tem mestu zapišite, komu se zahvaljujete za izdelavo diplomske naloge. Pazite, da ne boste koga pozabili. Utegnil vam bo zameriti. Temu se da izogniti tako, da celotno zahvalo izpustite.

Kazalo

Povzetek

Abstract

| | | |
|----------|---|-----------|
| 1 | Uvod | 1 |
| 2 | Pregled področja | 3 |
| 2.1 | Joga | 3 |
| 2.2 | Zgodovina joge | 4 |
| 2.3 | Asane in njihov pomen | 6 |
| 2.4 | Sorodne aplikacije | 8 |
| 3 | Načrtovanje aplikacije | 9 |
| 3.1 | Struktura mobilne aplikacije | 9 |
| 3.2 | Arhitektura aplikacije | 10 |
| 3.3 | Funkcionalnosti aplikacije | 11 |
| 4 | Razvoj mobilne aplikacije | 15 |
| 4.1 | Uporabljena tehnologija in orodja | 15 |
| 4.2 | Ogrodja | 15 |
| 4.3 | Orodja | 16 |
| 4.4 | Programski jeziki | 17 |
| 4.5 | Programi | 18 |
| 4.6 | Podatkovni nivo | 19 |
| 4.7 | Logični nivo | 20 |

| | |
|---|-----------|
| 4.8 Uporabniški nivo | 21 |
| 5 Predstavitev aplikacije | 25 |
| 5.1 Funkcionalnosti in delovanje aplikacije | 25 |
| 6 Sklepne ugotovitve | 35 |
| 6.1 Ideje za nadaljnji razvoj | 36 |
| Literatura | 37 |

Seznam uporabljenih kratic

| kratica | angleško | slovensko |
|---------|-----------------------------------|-----------------------------|
| IDE | integrated development enviroment | integrirano razvojno okolje |
| API | integrated development enviroment | integrirano razvojno okolje |

Povzetek

Naslov: Mobilna aplikacija Asana Master

Avtor: Klementina Garbajs

Asane oz. položaji imajo v jogi velik pomen, zato je pomembno, da je izvajanje le teh pravilno in v skladu z zmožnostmi posameznika. Prav z namenom učenja pravilne izvedbe položajev, spremeljanjem napredka in sledenja lastnim občutkom, smo razvili mobilno aplikacijo za Android, ki je primerna tako za začetne kot nadaljevalne praktikante, saj so Asane razdeljene na različne stopnje, prav tako pa si praktikant lahko sam določi svoje cilje, katerim lahko sledi, objavi sliko napredka in ustvarja zapiske oz. dnevnik občutkov. Aplikacija je razvita z uporabo ogrodja React Native in odprtokodno platformo Expo.

Ključne besede: Joga, mobilna aplikacija, Expo, React Native.

Abstract

Title: Asana Master mobile application

Author: Klementina Garbajs

For dissertation, we decided to develop the Asana Master mobile application, which would completely bypass the journey of individuals through his Yoga paths. Asanas or positions are of great importance in Yoga, so it is important to perform them correctly and in accordance with the abilities of individuals. In order to learn the correct execution of positions, monitor progress and the last last feeling of practitioners, we have developed an application that is suitable for both initial and advanced practices, as the Asanas are divided into different stages, and the practitioner can set their own goals.

Keywords: Yoga, mobile app, Expo, React Native.

Poglavlje 1

Uvod

Joga je za veliko ljudi le oblika sprostitve, način napajanja z energijo, vadba za boljše počutje. Za tiste, ki se z njo ukvarjajo bolj intenzivno, pa je predvsem izjemno, do potankosti izdelano in premišljeno urjenje osebnosti, ki lahko človeku omogoči, da dozori. Joga je pot k sebi. Nima pa joga le pozitivnih učinkov na um, temveč tudi na telo. Z leti postane naše telo manj gibčno, utrujeno, bolj občutljivo in nagnjeno k poškodbam, kar pa lahko učinkovito odpravimo z vadbo joge, saj telo postane prožnejše, bolj gibljivo in močnejše. Joga so telesni položaji ali asane preko katerih začutimo in spoznavamo svoje telo, ga nadzorujemo in izboljšujemo, vendar praksa asan ne sme biti brezglava in pretirana, zato je poznavanje položajev in pravilne izvedbe le-teh zelo pomembno. In kaj je asana? Sanskrtski izraz asana pomeni poza ali položaj telesa. Asana je psiho-somatska vaja za telo in um, saj z njo vplivamo tako na počutje telesa, kot tudi uma. Pozorni moramo biti tako na pravilen položaj telesa, kot tudi na naše dihanje.

Prav z namenom pravilnega izvajanja položajev in enostavnim spremeljanjem napredka, smo se za diplomsko nalogo odločili razviti Android mobilno aplikacijo Asana Master, ki bi bodočim joga učiteljem in tudi vsem ostalim omogočala lažje in bolj zabavno obliko učenja, ter dodatno motivacijo za napredek. Uporabnik ima možnost učenja o različnih položajih v jogi, možnost spremeljanja programa za izboljšanje gibljivosti, inverzij, itd., prav tako pa

ima možnost ob tem še spremljati svoj lasten napredek, si podati samooceno o lastnem napredku in izraziti svoje občutke preko zapiskov.

Aplikacija je razvita z uporabo ogrodja React Native in odprtokodno platformo Expo. Za pisanje kode so bili uporabljeni jeziki, Java Script, Type-Script in SQL. Aplikacija teče na Expovem lokalnem strežniku, za povezavo do baze podatkov pa skrbi strežnik Express. Podatki so shranjeni v bazi, ki je nameščena na lokalnem strežniku MySQL.

Poglavlje 2

Pregled področja

2.1 Joga

”Joga združuje načine zavedanja, zaznavanja, doživljanja, soočanja, spoznavanja in delovanja, preko katerih je človek zmožen bivati neodvisno od lastne pogojenosti” [6]. Beseda joga izhaja iz korena besede *yuj*, ki pomeni združiti, povezati. Nakazuje na to, da se s tehnikami joge osvobodimo vezi in omogočimo združitev duše z absolutnim zavedanjem. Skozi čas se je razvila vrsta raznih praks in zvrsti joge. Nekatere izmed praks imajo s klasično jogo vedno manj skupnega, saj je po tradiciji učenje joge zahtevalo preučevanje tekstov ter pridobivanje znanja s pomočjo guruja, dan danes pa je več poudarka na izvajanju položajev in na splošnem fizičnem aspektu joge. Obstaja več različnih poti joge, ki se med seboj dopolnjujejo. Posameznik izbere pot, ki je najbližja njegovemu trenutnemu stanju zavedanja. Glavne poti so štiri: Karma joga (pot nesebičnega delovanja za dobro drugih in sveta), Jnana joga (pot modrosti in intelektualnega raziskovanja), Bhakti joga (pot predanosti in ljubezni do izbranega idealja) in Raja joga (kraljevska pot joge, ki jo sestavlja osem krakov, skozi katere se poda posameznik na poti do samorealizacije). Poleg naštetih poti obstajajo še nekatere druge, med njimi je tudi Hatha joga, katero najpogosteje srečamo na zahodu, saj daje povzornost predvsem izvajanju asan in pranajame.

2.2 Zgodovina joge

Zibelka joge naj bi bila indska civilizacija, nekje v letih od 2600 do 1900 pr. n. št., ko je joga kot posebna metoda preseganja človeške pogojenosti, omenjena v vedskih spisih, ki so nastali na območju današnje Indije [12]. Preden se je joga uveljavila kot osebna praksa, so kot prevladujoči načini samo-preseganja služili obredni rituali žrtvovanja. Preko teh ritualov so si ljudje skušali zagotoviti moč nadzora nad lastno minljivostjo. S prihodom nomadskih plemen na območje indske civilizacije pa so obredni rituali začeli izgubljati svojo vlogo in začelo se je vedsko obdobje, ki je trajalo nekje od leta 1700 do 1100 pr. n. št. V tem času se je znanje o jogi prenašalo preko ustnega izročila, šele na začetku 2. tisočletja n. š. pa je bilo znanje zapisano v Vede. Vede veljajo za ene najstarejših besedil sveta in predstavljajo osrednje bistvo duhovnosti, hinduizma in temelje joge.

S koncem Vedskega obdobja teče začetek Asketskih praks, ki so izvor jogijskih praks. Asketi so popolnoma zavračili avtoriteto Ved in pomembnost ritualov. To odobje je trajalo od 6. do 3. stol. pr. n. št., za tem pa je sledilo obdobje Upanišad. Zmožnost odrešitve je v tem času prenehala obstajati kot privilegij pogoj s spoznavanjem ritualnih praks, saj se je pozornost preusmerila na osebi in njen način bivanja. Rishiji - modreci in avtorji upanišadskih spisov so začeli poudarjati, da je odrešitev zmožen doseči vsak z načinom svojega bivanja in delovanja. To bivanje in delovanje so opisali v šamkhyah”, iz katerih izhaja tudi teoretični princip joge. Ta temelji na predpostavki, da je človeško trpljenje posledia nepoznavanja sebe in vzrokov svojih dejanj. Zaradi neznanja se Sebstvo identificira s fizičnim telesom. Rešitev je možna šele, ko se Sebstvo osvobodi teh napačnih poistovetenj in materialnih vezi. S tem, ko je možen bivati onstran fizičnega in psihičnega izkustva, lahko spoznava in spreminja pogoje bivanja in vzroke svojega delovanja.

Filozofija joge je prevzela osnovne predpostavke šamkhye”, le da odrešitve

ne povezuje toliko s spoznanjem, kolikor z načinom delovanja. Človek je zmožen preseči lastno pogojenost in bivati neodvisno od materialnega sveta (za katerega se je predpostavlja, da vsebuje tako fizično kot psihično realnost) prav z določenim načinom doživljanja in delovanja v materialnem svetu. Omenjen način doživljanja in delovanja je v filozofiji joge opisan kot jogijska praksa.

Prvo sistematično zbirko jogijskih praks, ki so se izvajale na območju današnje Indije je izdelal Patanjali v svojih Yoga-sutrah, prvem samostojnjem besedilu o jogi, v katerih avtor opredeli osmeročleno pot joge [7]. Besedilo je eno glavnih jogijskih besedil na katerega se lahko opre novodoben praktikant joge. V njih Patanjali predstavi skupek navodil, ki jih mora praktikant izvajati istočasno in v celoti, če želi napredovati po poti joge. Vrednote etike in morale praktikant skrbno in počasi integrira v svoje življenje.

Tradicionalno jogo sestavlja osem temeljnih vej Raja joge (ash – osem, anga – veja) in obsegajo vse vidike telesnega, čustvenega in mentalnega življenja. Osem vej je sistematično opisanih v Yoga-sutrah:

1. Yama ali etične zmožnosti: navodila kako naj jogi deluje v odnosu do sveta s poudarkom na samodisciplino in vzdržnost.

- nenasilje (ahimsa) – nezmožnost škodovati sebi in drugim
- resnicoljubnost (satya) – usklajenost govora z dejanji
- onemogočanje kraje oziroma težnje po prilaščanju tega kar nam ne pripada (asteya)
- neodvisnost od spolnih potreb (brahmaccarya)
- onemogočenje pohlepa (aparigraha) oziroma težnje po prilaščanju tega česar ne potrebujemo

2. Niyama ali discipline: navodila kako naj jogi deluje do sebe

- čistost telesa in duha (sauca)
- notranje zadovoljstvo ali zmožnost izkusiti zadovoljstvo neodvisno od zunanjih dejavnikov (santosha)

- razvoj samodispline (tapas)
- odsotnost besed (kastha mauna)
- odsotnost gest in kretenj (akara mauna)
- samo-preučevanje (svadhyaya)
- predanost božanskemu (ishvara pranidana)

3. Asana ali telesni položaji

4. Pranayama ali tehnike nadzora energije

5. Pratyahara ali neodvisnost čutil od zaznavanja zunanjih dejavnikov

6. Dharana ali zbranost in zmožnost nadzorovanega usmerjanja pozornosti na en predmet

7. Dhyana ali meditacija oz. podaljšana koncentracija

8. Samadhi ali stanje skupne zavesti

2.3 Asane in njihov pomen

Beseda asana pomeni stabilen in udoben položaj. Največkrat pa jo v slovenščini enačimo s terminom vaja ali položaj.

Praktikant želi najti optimalno naravnavo znotraj asane glede na posebnosti in sposobnosti svojega telesa. Svojo zaznavo najprej osredotoči na občutenje položaja posameznega dela telesa, ko najde varen položaj, pa v njem išče udobje in stabilnost ter poskuša zmanjšati telesni napor, da bi lahko sčasoma njegova pozornost prešla od fizične zaznave.

V hatha jogi so asane ena osnovnih tehnik telesnega gibanja [2]. Asane dajejo telesu gibljivost in pravilno držo, ki je temelj pravilne izvedbe jogijskih položajev. Z njo med vadbo preprečimo morebitne mišične ali skeletne poškodbe in krepimo mišice stabilizatorje trupa. Za boljše življenje in

učinkovito prakso brez bolečin, se moramo naučiti zavestnega nadzora telesne drže in zavedanja poravnave telesa v vsakem položaju.

Jogajska praksa od praktikantov ne zahteva le pravilnega izvajanja asan temveč tudi pozornost in usmerjenost uma v to, kar počnemo. V asani smo, kot pove že sanskrtsko ime ”as” oz. biti, tukaj in zdaj.

Nekateri viri navajajo, da naj bi bilo asan nekje med 70 in 80 tisoč. Poimenovali so jih po živalih, rastlinah, mineralih in božanstvih. Večina asan vsebuje hatha princip, ki pomeni princip moči. Zelo pomembno je, da asane izvajamo primerno svojim telesnim sposobnostim in pravilno, kar pomeni postopnost učenja in poslušanje svojega telesa. Nikoli ne izvajamo asan preko svojih zmožnosti, ampak le do praga bolečine.

Izvedba asane naj bi bila stabilna in hkrati selektivno sproščena. Dihanje je sproščeno, enakomerno, pozornost je usmerjena v počutje telesa, opazujemo sporočila telesa, umirimo um in zadržujemo položaj toliko časa, dokler čutimo asano v vseh navedenih lastnostih.

Skozi asane se nam razkrivajo naši ustaljeni psihofizični vzorci, ki jih običajno doživimo kot omejitve. V jogi ni tekmovalnosti, vsak je v svoji asani, ki trenutno ustreza njegovemu psihofizičnemu stanju. Učitelji joge pogosto pravijo, da je tista asana, ki nam je najmanj všeč, pravzaprav naša asana.

Z rednim in discipliniranim izvajanjem asan se prožnost telesa izredno poveča. Mnoge asane učinkujejo tudi terapevtsko in lahko odpravijo težave, ki nas pestijo dolgo časa in so morda videti nerešljive, odpravi pa jih lahko že pravilna drža telesa. Hkrati se poveča tudi naša prožnost duha: opazimo lahko spremembe v svojem dojemanju sveta, samozavesti, odprtosti, sposobnosti, da ocenimo položaj in ugotovimo, kaj je bistvo in kaj nepomembna navlaka. Trdno stojimo na tleh, z odprtim srcem in notranjo radostjo, ki ne potrebuje zunanjih dražljajev. Takrat smo prestopili prvo, pomembno stopnico na poti joge. Bistvo izvajanja asane ni nujno v popolni izvedbi. Bistvo asane je v popolnem medsebojnem zavedanju duše in telesa.

2.4 Sorodne aplikacije

Pri pregledu sorodnih aplikacij smo ugotovili, da na trgu obstaja nekaj rešitev za izvajanje joge v angleškem jeziku in nekaj slovenskih prevodov.

Ena izmed boljših aplikacij, ki jo trenutno uporabljamo, je aplikacija Track Yoga [16], kjer ima uporabnik možnost pregleda nad različnimi asanami, prikazanimi tudi z videoposnetki, prav tako pa ima na voljo različno dolge posnetke vodene prakse oz. možnost sledenja programu, npr. programu za izboljšanje fleksibilnosti, zmanjševanju stresa, meditacija in drugo. V aplikaciji ima vsak uporabnik tudi svoj profil, kjer spremlja koliko programov in vodenih vadb je že opravil.

Ena izmed aplikacij, ki so osredotočene bolj na učenje posameznika o asanah, pa je aplikacija Yoga 108 [21]. V aplikaciji uporabniku ponujajo pregled nad različnimi asanami, njihovimi izgovorjavami in angleškimi prevodi, vendar pa ne ponujajo podrobnega opisa asane, njenih prednosti in prikaza pravilne izvedbe z videoposnetkom.

Obstaja seveda tudi veliko drugih aplikacij, ki pa so vse bolj ali manj osredotočene le na to, da uporabnika vodijo pri vadbi joge in ponujajo različne programe z videoposnetki, katerim potem posameznik sledi in prakticira jogo, a lahko uporabnik te aplikacije le pasivno uporablja. Zgoraj naštete aplikacije nimajo možnosti delanja zapiskov in ne omogočajo spremljanja lastnega napredka, kot je objavljanje slik, podajanje samoocene in interaktivno sledenje programu, kar uporabnika še bolj motivira.

Poglavlje 3

Načrtovanje aplikacije

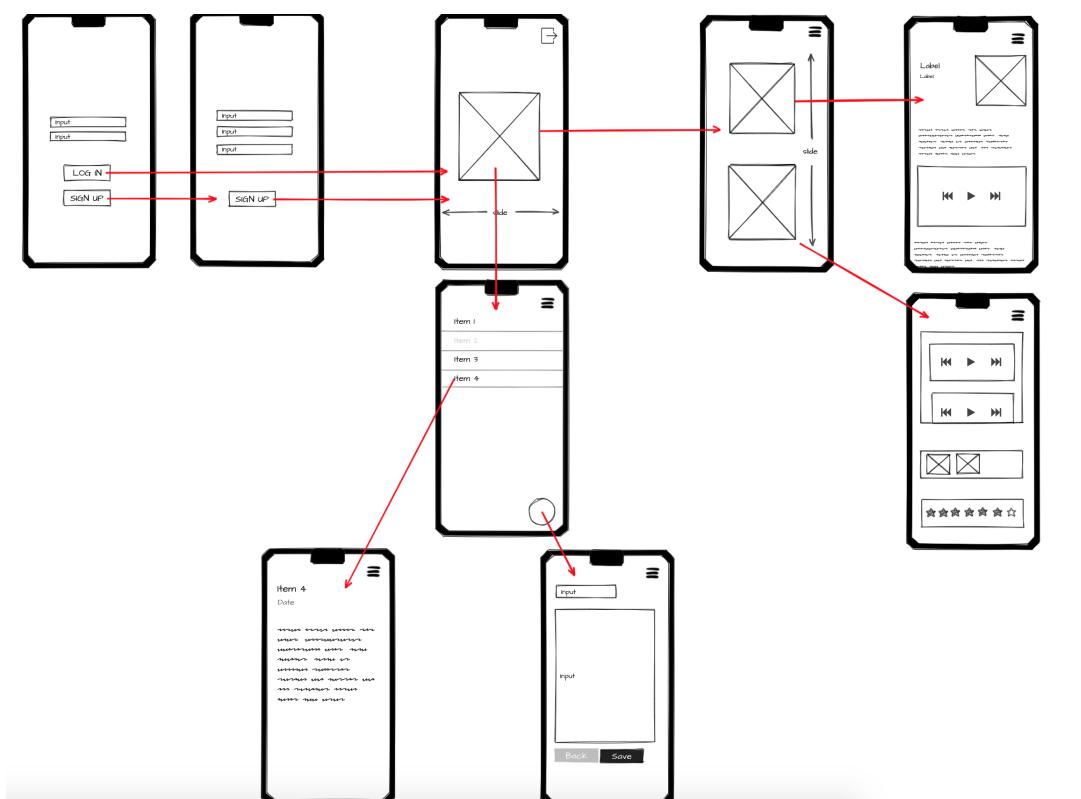
Razvoj mobilnih aplikacij ne bi smel biti več tako osredotočen le na eno glavno funkcionalnost, saj počasi prihaja čas, ko smo ljudje vedno bolj zasičeni z različnimi podatki, aplikacijami in načini shranjevanja. Vedno bolj se bomo nagibali k aplikacijam, ki omogočajo večje število funkcionalnosti v enem. Z namenom, da ima uporabnik, vse kar potrebuje za sledenje lastnemu napredku, na enem mestu, smo se odločili za izdelavo mobilne aplikacije, ki bi to tudi omogočala.

3.1 Struktura mobilne aplikacije

Načrtovanje mobilne aplikacije se je začelo z izdelavo načrta strukture mobilne aplikacije, katerega osnovni namen je začrtati osnovno postavitev vsebine (struktura, elementi na strani), še preden sledi oblikovanje, (Slika 3.1). V načrtu smo zasnovali zaslon za Prijavo in Registracijo, strukturo glavnega menija ter različnih podmenijev, ki se odprejo s klikom na element v glavnem meniju. Glavni strukturi podmenijev sta na sliki prikazani s puščicami iz trete skice (desno in navzdol). Skice desno predstavljajo strukturo podmenija Asanas in Goals, skice spodaj pa strukturo podmenija Notes.

Struktura ne vsebuje podrobnosti oblikovanja – barv, tipografij, slik. Lahko se skicira zelo na hitro na listu papirja, ali pa se pripravi v digi-

talni obliki s pomočjo različnih programov oz. aplikacij. Za izdelavo orisa strukture je bila uporabljena spletna platforma WireframePro [20], ki je vodilno orodje za oblikovanje načrtov uporabniškega vmesnika za spletna mesta in mobilne aplikacije. Vključuje že pripravljene komponente uporabniškega vmesnika za iOS, Android, splet, Bootstrap, WatchOS,... in ima tudi obsežno knjižnico predlog, ki uporabnikom olajša izdelavo načrtov.



Slika 3.1: Načrt strukture mobilne aplikacije.

3.2 Arhitektura aplikacije

Arhitektura aplikacije je sestavljena iz treh glavnih nivojev: predstavitvenega nivoja, logičnega nivoja in podatkovnega nivoja, (Slika 3.2). V predstavi-

tvenem nivoju se nahaja grafični vmesnik oz. Android mobilna aplikacija, ki zahteva podatke iz logičnega nivoja. V logičnem nivoju se nahaja zaledni sistem (REST API), ki procesira zahteveke iz predstavitevnega nivoja in komunicira s podatkovno bazo na podatkovnem nivoju. Prednosti take arhitekture sta bolj enostavno enostavno ponovno uporabljanje dela aplikacije in integriteta podatkov [1]. Slabost pa je povečana kompleksnost dela.



Slika 3.2: 3-nivojska arhitektura aplikacije.

3.3 Funkcionalnosti aplikacije

Glavne funkcionalnosti, ki jih ima uporabnik na voljo v aplikaciji Asana Master, so razdeljene in opisane po posameznih podmenijih v aplikaciji:

- **Prijava in registracija:**
 - Registracija: kreiranje novega uporabniškega računa
 - Prijava: prijava v mobilno aplikacijo z uporabniškim računom
- **Asane:**
 - Ogled položajev/asan: ogled položajev oziroma asan

- Ogled videoposnetkov: ogled videoposnetkov položajec v mobilni aplikaciji

Podatki o asanah in jogi so vzeti in prepisani v podatkovno bazo, iz knjige The Top 100 Best Yoga Poses Relieve Stress, Increase Flexibility, and Gain Strength, avtorice Susan Hollister [8]. Vir nekaterih slik je tudi zgoraj omenjena knjiga, nekatere pa so bile pridobljene preko spletne strani Verywellfit [19]. Vir videoposnetkov je portal YouTube. Posnetek se predvaja neposredno v sami aplikaciji, brez dodatnega odpiranja videoposnetka izven mobilne aplikacije. Za to funkcionalnost je uporabljena React Native knjižnica react-native-youtube-iframe. Knjižnica omogoča predvajanje posnetka skozi celoten ekran mobilne naprave, s klikom na naslov posnetka pa uporabnika preusmeri na vir videoposnetka na Youtubu, kjer si ga lahko ogleda izven aplikacije.

- **Goals:**

- Ocenjevanje napredka: ocenjevanje lastnega napredka uporabnika
- Nalaganje slik: nalaganje slik napredka uporabnika
- Urejanje slik: urejanje slik uporabnika
- Ogled slik: ogled slik napredka uporabnika
- Ogled videoposnetkov programa v mobilni aplikaciji

Za nalaganje slik je uporabljena Expova knjižnica expo-image-picker, ki omogoča dostop do uporabniškega vmesnika sistema za izbiranje slik in videoposnetkov iz knjižnjice telefona ali fotografiranje s fotoaparatom. Knjižnica omogoča tudi obračanje, obrezovanje in zrcaljenje slike. Za ogled naloženih slik pa je uporabljena React Native knjižnica react-native-image-zoom-viewer, ki omogoča jasen in enostaven pregled slik, z dodatno možnostjo povečave slike. Za podajanje ocene napredka je uporabljena knjižnica react-native-ratings, ki omogoča enostavno podajanje ocen v obliki zvezdic.

- **Notes:**

- Kreiranje novega zapiska: uporabnik lahko kreira nov zapisek
- Shranjevanje novega zapiska: uporabnik lahko shrani novo ustvarjen zapisek
- Brisanje zapiska: uporabnik lahko željeni zapisek izbriše
- Ogled zapiskov: uporabnik si lahko ogleda ustvarjene zapiske

Zapiski so razdeljeni v kategorije ciljev, ter razvrščeni po dnevih. Zapiski, ki so dodani preko gumba za dodajanje zapiskov v podmeniju Notes, se shranijo pod kategorijo Random, saj niso vezani na nobenega izmed ciljev.

Poglavlje 4

Razvoj mobilne aplikacije

4.1 Uporabljena tehnologija in orodja

Mobilna aplikacija je razvita z uporabo ogrodja React Native in ogrodja Expo, kar omogoča izdelavo mobilnih aplikacij tako za operacijski sistem Android kot iOS. Za pisanje kode so uporabljeni jeziki TypeScript, JavaScript in SQL, koda pa je napisana v programu Visual Studio Code. Za shranjevanje podatkov, je uporabljena relacijska podatkovna baza MySql, ki je povezana z lokalnim strežnikom. Podatki so shranjeni v programu Table-Plus, ki omogoča enostaven vnos ter urejanje podatkov. Za prikaz podatkov iz podatkovne baze v uporabniški vmesnik skrbi Expressov API server. Testiranje aplikacije poteka preko Expovega orodja za testiranje Expo Developer Tools. Vse spremembe v kodi, ter zgodovina spremenjenih verzij aplikacije so shranjene v programu Git.

4.2 Ogorodja

- **React Native:** React Native je JavaScript ogrodje za pisanje izvornih mobilnih aplikacij za operacijska sistema iOS in Android. Temelji na Reactu, Facebookovi JavaScript knjižnici za gradnjo uporabniških vmesnikov, a namesto na brskalnik cilja na mobilne platforme [13]. Re-

act Native, je Facebook prvič izdal kot odprtakodni projekt leta 2015. V samo nekaj letih je postal ena najboljših rešitev za mobilni razvoj. React Native razvoj se uporablja za pogon nekaterih vodilnih svetovnih mobilnih aplikacij, vključno z Instagramom, Facebookom in Skypeom.

- **Expo:** Expo je ogrodje, ki se uporablja za izdelavo React Native aplikacij. Gre za nabor orodij in storitev, zgrajenih okoli React Native-a in izvornih platform, ki nam pomagajo razvijati, graditi, uvajati in hitro iterirati v iOS, Android in spletne aplikacije iz iste kodne baze JavaScript / TypeScript [3]. Z uporabo Expo-ta ne potrebujemo znanja o izvorni iOS ali Android kodi, kar pomeni, da tudi uporaba orodij kot sta Xcode ali Android Studio, ni potrebna. Njegov cilj je omogočiti hiter razvoj, ne da bi morali veliko časa porabiti za vzpostavljanje razvojnega okolja. Expo ne zahteva nastavitev IDE-jev, specifičnih za platforme, kot je to v primeru React Native CLI, in ke zato celoten postopek namestitve veliko bolj enostaven. Expo prav tako ponuja okolje za testiranje Expo Developer Tools, ki omogoča enostavno testiranje aplikacij, tako na spletnem vmesniku, mobilnem simulatorju, kot tudi na mobilni napravi.
- **Express:** Express je minimalno in prilagodljivo ogrodje spletnih aplikacij Node.js, ki ponuja robusten nabor funkcij tako za spletne, kot mobilne aplikacije [4]. Z neštetimi metodami pripomočkov HTTP in vmesno programsko opremo, ki jih imamo na voljo, je omogočeno enostavno in hitro ustvarjanje robustnega API -ja.

4.3 Orodja

- **npm:** Npm je privzeti upravitelj paketov za izvajalno okolje JavaScript Node.js Namešča module, tako da jih vozlišče lahko najde, in inteligentno obvladuje konflikte med odvisnostmi [11]. Sestavljen je iz odjemalca ukazne vrstice, imenovanega tudi npm in spletne baze javnih

in plačljivih zasebnih paketov, imenovane register npm. Izjemno prilagodljiv je za podporo najrazličnejšim primerom uporabe. Najpogosteje se uporablja za objavljanje, odkrivanje, nameščanje in razvoj vozliščnih programov.

4.4 Programski jeziki

- **Java Script:** JavaScript (krajše tudi JS) je skriptni programski jezik, katerega je razvil Netscape, da bi bil v pomoč programerjem pri izdelavi spletnih strani. Večina tistih, ki pozna vsaj nekaj osnov programiranja, ko zasliši ime JavaScript, hitro poveže s programskega jezika Java, s katerim imata kar nekaj skupnih lastnosti [9]. Pri skriptnih jezikih v splošnem obdelava kode traja dalje kot pri zbirnih jezikih, vendar pa so veliko bolj uporabni pri krajsih programih zaradi njihove enostavnosti. Uporabnost JavaScript-a pride najbolj do izraza pri razvoju dinamičnih spletnih strani in dodajanju interaktivnosti na želeno stran.
- **TypeScript:** TypeScript je odprtokodni jezik, ki temelji na JavaScriptu, enem izmed najbolj uporabljenih jezikov na svetu, z dodajanjem statičnih definicij tipa [17]. Tipi nam ponujajo način za opis oblike predmeta, zagotavljajo boljšo dokumentacijo in omogočajo TypeScriptu, da preveri, ali koda deluje pravilno. Vsa veljavna koda napisana v jeziku JavaScript je tudi koda TypeScript. TypeScript se prek prevajalnika TypeScript ali Babel pretvori v JavaScript. JavaScript koda je čista in preprosta in se izvaja kjer koli, kjer deluje JavaScript: v brskalniku, na Node.js ali v aplikacijah.
- **SQL:** SQL je najbolj razširjen računalniški jezik, ki omogoča kreiranje, sprememjanje, branje in manipulacijo s podatki, shranjenimi v relacijski PB [14]. Širitev jezika gre v smeri podpore objektno relacijskim podatkovnim bazam. Jezik SQL je standardiziran (ANSI / ISO), vendar je upoštevanje standardov s strani nekaterih proizvajalcev SUPB

vprašljivo. Njegove značilnosti so, da je zasnovan na množicah (ker izhaja iz relacijskega modela podatkov), ter enostaven nabor ukazov.

4.5 Programi

- **Visual Studio Code:** Za pisanje kode je uporabljen program Visual Studio Code, znan tudi kot VS Code, ki je Microsoftov brezplačen odprtokodni urejevalnik besedil. VS Code je na voljo za Windows, Linux in macOS [18]. Čeprav je urejevalnik razmeroma lahek, vključuje nekaj zmogljivih funkcij, zaradi katerih je VS Code v zadnjem času postaleno najbolj priljubljenih orodij za razvojno okolje, saj podpira široko paleto programskih jezikov od Java, C ++ in Pythona do CSS, Go in Dockerfile. Poleg tega VS Code omogoča dodajanje in celo ustvarjanje novih razširitev, vključno s kodami, iskalniki napak in podporo za razvoj v oblaku in spletnem razvoju.
- **MySQL:** MySQL je odprtokodni sistem za upravljanje relacijskih baz podatkov [10]. Kot del široko uporabljenega svežnja tehnologij LAMP (ki ga sestavlja operacijski sistem, ki temelji na Linuxu, spletni strežnik Apache, baza podatkov MySQL in PHP za obdelavo) se uporablja za shranjevanje in pridobivanje podatkov v najrazličnejših priljubljenih aplikacijah, spletnih mestih in storitvah.
- **TablePlus:** Za pregledovanje in urejanje baze podatkov je bil uporabljen program TablePlus [15]. TablePlus je sodobna, izvorna aplikacija s čistim uporabniškim vmesnikom, ki razvijalcem omogoča sočasno upravljanje podatkovnih baz na zelo hiter in varen način. TablePlus podpira večino priljubljenih baz podatkov, kot so MySQL, Postgres, SQL Server, SQLite, Microsoft SQL Server, Redis, Redshift, Oracle in še veliko več.
- **Git:** Git je brezplačen in odprtokodni porazdeljeni sistem za nadzor različic, zasnovan za hitro in učinkovito obdelavo vsega, od majhnih do

zelo velikih projektov [5].

4.6 Podatkovni nivo

Podatkovna baza je zbirka povezanih podatkov. Prednosti uporabe podatkovne baze so možnost shranjevanja večjih količin podatkov, hitrejši prenos podatkov in hitrejša obdelava podatkov, (Slika 4.1). Pri podatkovnem modeliranju se izvaja abstrakcija in poenostavitev nekega segmenta realnega sveta. Na osnovi modela se potem izdela podatkovna baza. Podatkovni model je povezana zbirka konceptov namenjenih opisovanju podatkov in opravljanju s podatki, ki opredeljuje sintaksco in semantiko stavkov formalnega jezika. Za predstavljeno podatkovno bazo je bil uporabljen entitetno relacijski podatkovni model, ki je sestavljen iz entitet, atributov in relacij. Entiteta je v podatkovni bazi predstavljena z množico atributov. Atribut je podatkovni element, s katerim opišemo lastnost entitete. Za definiranje povezav med dvema ali več entitetami so uporabljene relacije, ki lahko pripradajo različnim entitetnim množicam. Spodaj je predstavljen fizični model s tabelami in njihovimi relacijami. Podatkovna baza je sestavljena iz petih tabel: *Users, Asanas, Goals, Images, Notes*.

Entitete in njihovi atributi

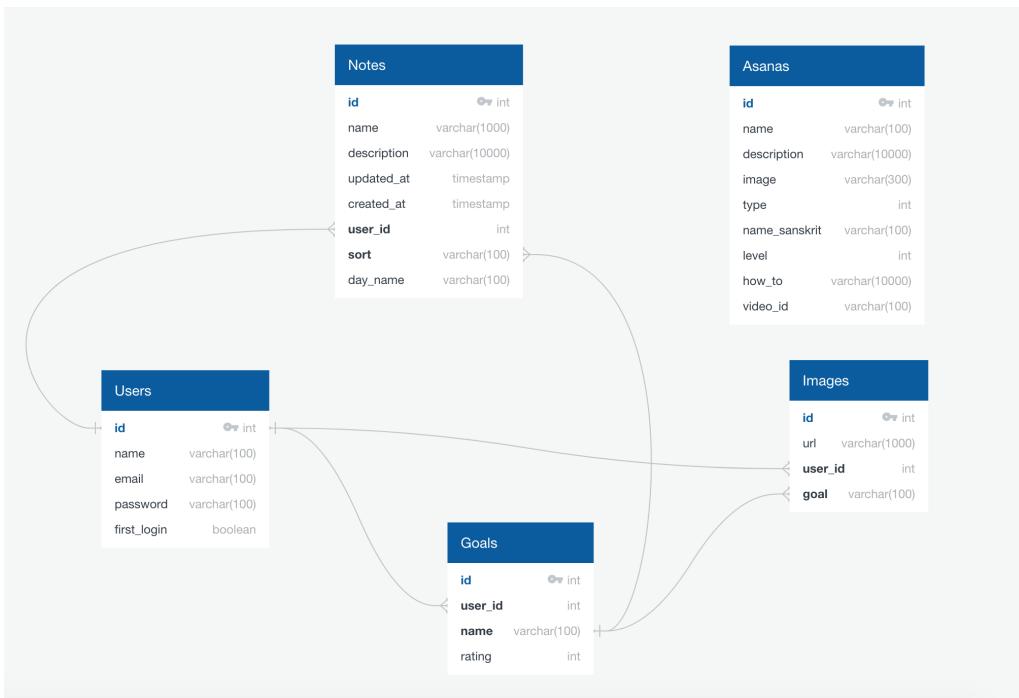
- Users: id (primarni ključ), name, email, password, first_login.
- Asanas: id (primarni ključ), name, description, image, type, name_sanskrit, level, how_to, video_id.

V tabeli Asanas imamo pri vsakem položaju shranjeno njegovo stopnjo težavnosti (atribut level), kar nam omogoča enostavno sortiranje v omenjene težavnosti.

- Goals: id (primarni ključ), user_id (tuji ključ), name (tuji ključ), rating.
- Images: id (primarni ključ), url, user_id (tuji ključ), goal (tuji ključ).

V tabeli Images so slike shranjene lokalno na disku, v bazi pa se nahaja le referenca na datoteko (atribut url).

- Notes: id (primarni ključ), user_id (tuji ključ), sort (tuji ključ), name, description, updated_at, created_at, day_name.



Slika 4.1: Fizični model podatkovne baze.

4.7 Logični nivo

Logični nivo predstavlja REST API. Za njegov ravoj je bilo uporabljeno ogrodje Express.

API je v aplikaciji posrednik med podatkovno bazo in uporabniki aplikacije. Sprejema zahtevke s strani uporabe aplikacije in izvaja ustrezne operacije na podatkovni bazi. Uporabljeni so bili naslednji zahtevki HTTP za različne namene:

```

server > TS server.ts ...
1 import {Errback, NextFunction, Request, Response} from 'express';
2 import {asanamanager} from './asanamanager';
3 import {notesManager} from './notesManager';
4 import {registrationManager} from './registrationManager';
5 import {imagesManager} from './imagesManager';
6
7 const express = require('express');
8 const bodyParser = require('body-parser');
9 const app = express();
10 var cors = require('cors');
11
12 const port = process.env.PORT || 8080;
13
14 app.listen(port);
15
16 console.log(`Server listening on ${port}`);
17
18 app.use(express.json());
19 app.use(bodyParser.json());
20 app.use(cors());
21 app.use(bodyParser.urlencoded({ extended: true }));
22
23 app.get('/', (req: Request, res: Response) => res.status(301).redirect('/api'));
24 app.get('/api', (req: Request, res: Response) => res.status(200).send({
25   message: 'Welcome to Express API',
26 }),);
27
28 app.all('/asanas/:api', asanaManager);
29 app.all('/notes/:api', notesManager);
30 app.all('/register/:api', registrationManager);
31 app.all('/images/:api', imagesManager);
32
33 // Throw error when user enters wrong Endpoints
34 app.use((req: Request, res: Response) => res.status(404).send({error: 'Oops! Endpoint not found, Please Check that you are entering the right thing!' }));
35 app.use((err: Errback, req: Request, res: Response, next: NextFunction) => {res.status(500).send({
36   error: 'Invalid Request! Please Check that you are entering the right thing!',
37 });});
38

```

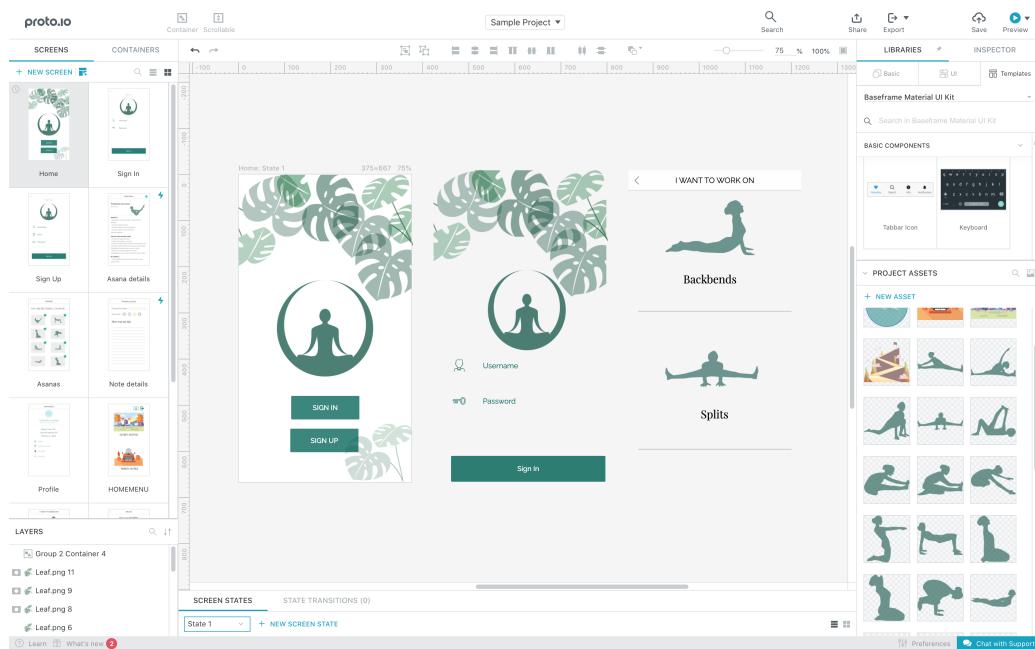
Slika 4.2: Koda za vzpostavitev REST API serverja.

- **GET:** za pridobivanje podatkov o asanah, zapiske, slike in ostale vnose v podatkovni bazi
- **POST:** za dodajanje različnih vnosov v podatkovno bazo, kot je dodajanje slik ter zapiskov
- **DELETE:** za brisanje vnosov v podatkovni bazi

4.8 Uporabniški nivo

Uporabniški nivo predstavlja Android mobilna aplikacija. Za njen razvoj je bilo uporabljeno ogrodje Expo z React Native-om. Aplikacija s pošiljanjem zahtevkov HTTP komunicira z APIjem in od njega prejema odgovore, nato pa ustrezno posodobi uporabniški vmesnik. Izdelava same mobilne aplikacije se je začela z izdelavo prototipa uporabniškega vmesnika mobilne aplikacije, saj je izdelava dobrega uporabniškega vmesnika pomemben sestavni del obli-

kovanja uporabniške izkušnje. Vmesnik je narejen na bolj pregleden način, da je tako bolj enostaven za uporabo. Prav s tem namenom so v aplikacijo dodane tudi številne slike in videoposnetki. Prototip je bil izdelan s pomočjo spletne platforme za izdelavo prototipov Proto.io, (Slika 4.3). Izdelani so bili nekateri ključni zasloni aplikacije, glavne funkcionalnosti aplikacije, povezave med njimi, oblikovano glavno ozadje aplikacije na začetnem prijavnem zaslonu ter zaslonu za registracijo in oblikovan glavni meni aplikacije.



Slika 4.3: Spletna platforma za ustvarjanje prototipov, Protoio.

Izdelavi prototipa je sledila raziskava tehnologij in primernih orodij za sam razvoj mobilne aplikacije, pri čemer je bila v ospredju primerjava različnih orodij za izdelavo mobilne aplikacije. Primerjana sta bila integrirano razvojno okolje (angl. integrated development environment, IDE) Android Studio za razvoj aplikacij za operacijski sistem Android, ter razvojno okolje Expo, ki z uporabo React Native platforme in orodij, omogoča razvoj tako spletnih, kot mobilnih aplikacij na operacijskem sistemu Android in iOS. Zaradi možnosti razvoja na obeh operacijskih sistemih ter predhodnih izkušenj,

je bilo nazadnje, za razvoj mobilne aplikacije, izbrano razvojno okolje Expo.

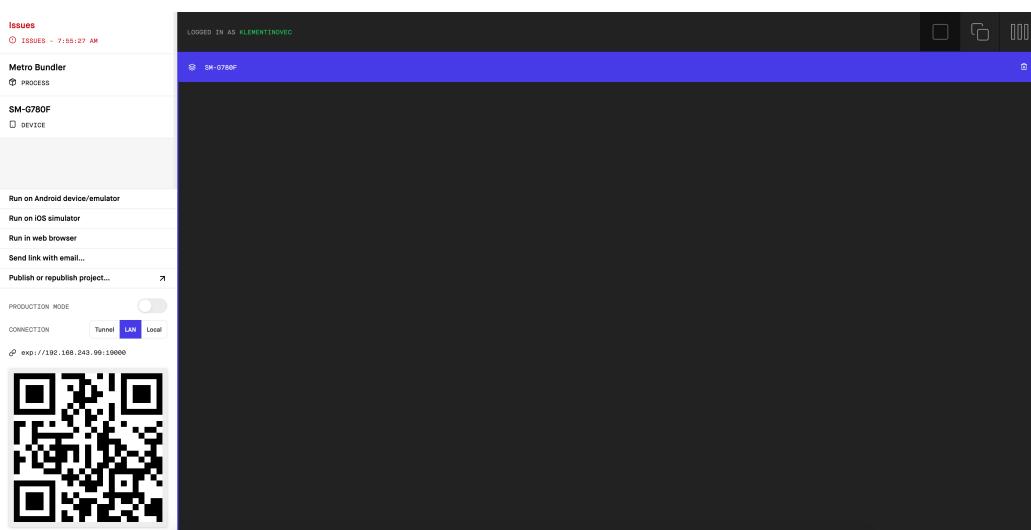
Po začetnem načrtovanju prototipa aplikacije je sledilo še zbiranje podatkov o jogi in položajih v jogi.

Poglavlje 5

Predstavitev aplikacije

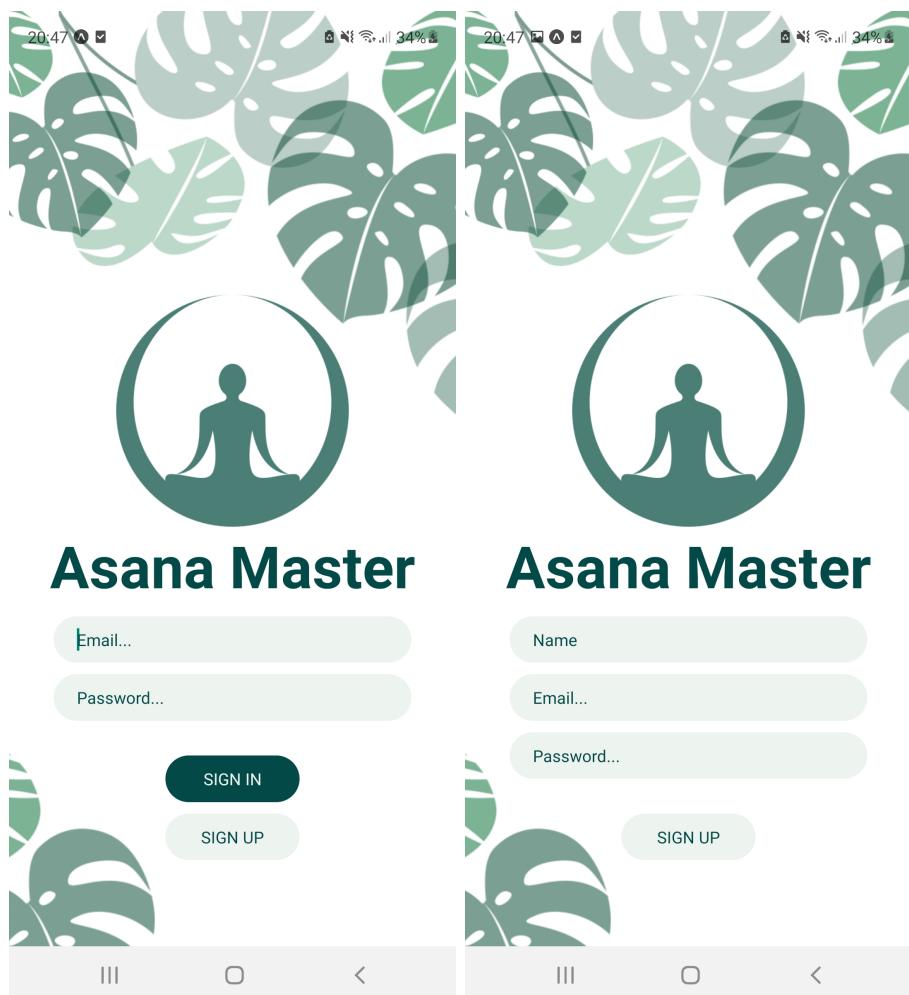
5.1 Funkcionalnosti in delovanje aplikacije

Za zagon aplikacije je potrebno najprej vzpostaviti povezavo s strežnikom z ukazom ”npm run server”, nato pa se z ukazom ”npm web buildžažene testno okolje Expo Developer Tools. V testnem okolju se nahaja možnost testiranja mobilne aplikacije na Android emulatorju, lahko pa se aplikacija testira preko mobilne naprave s skeniranjem QR kode, (Slika 5.1).



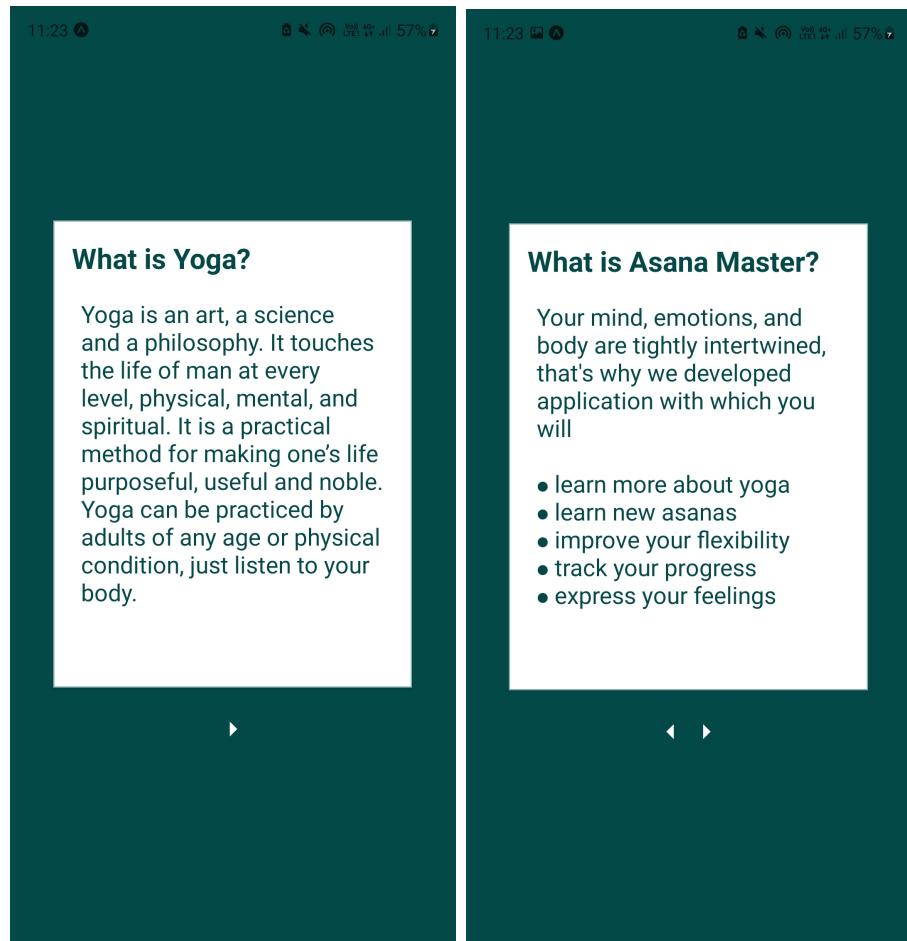
Slika 5.1: Expo Developer Tools testno okolje na spletnem vmesniku.

Ob zagonu mobilne aplikacije je uporabnik najprej preusmerjen na stran za prijavo, (Slika 5.1, levo). Uporabnik, ki že ima uporabniško ime in geslo, se lahko s klikom na Sign in enostavno prijavi v aplikacijo z že obstoječimi podatki. V kolikor uporabnik še ni registriran, mora najprej opraviti registracijo, s klikom na gumb Sign up, kar ga preusmeri na stran za registracijo, (Slika 5.1, desno). Za uspešno registracijo mora vnesti svoje ime, svoj elektronski naslov in geslo, ki ga bo uporabljal za prijavo. Ob registraciji uporabnika se v podatkovno bazo shrani nov uporabnik z vnesenimi podatki in sproži se prijava uporabnika.



Slika 5.2: Obrazec za prijavo (levo) in registracijo (desno)

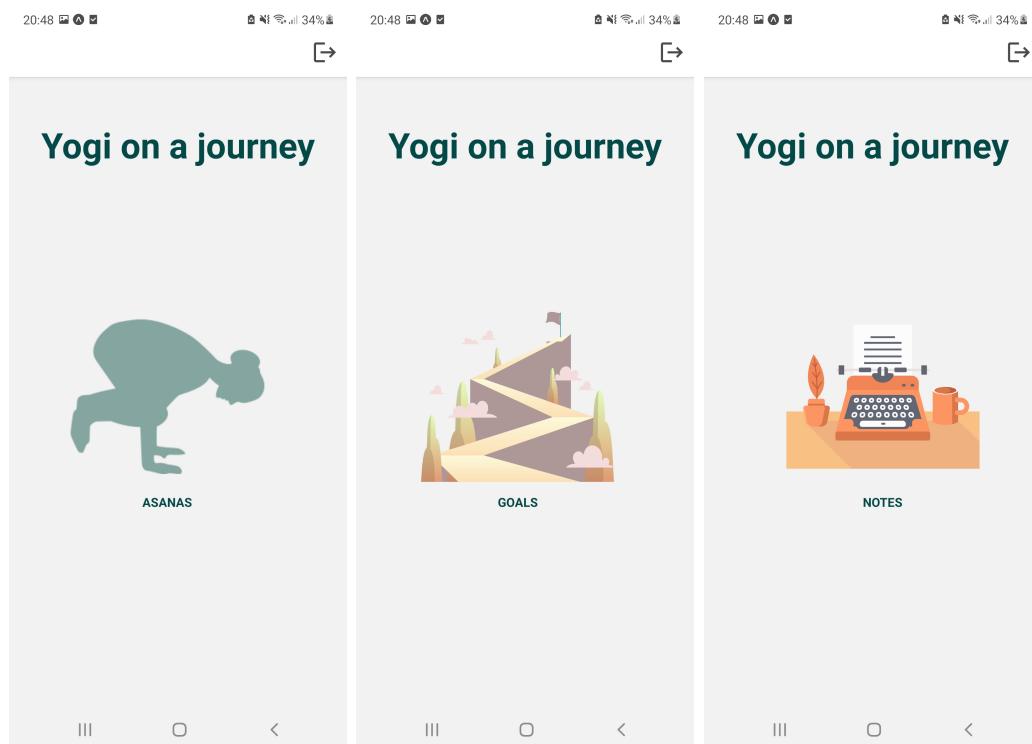
Ob prvi prijavi, uporabnika ob vstopu v aplikacijo pozdravi kratek uvod v aplikacijo, (Slika 5.1). Pozdrav je predstavljen na dveh straneh, med katerima se lahko uporabnik poljubno premika. Na prvi strani se uporabnik seznaní s pojmom joga in na drugi strani s kratkim opisom aplikacije.



Slika 5.3: Prva in druga stran začetnega pozdrava, ob prvi prijavi uporabnika.

Po začetnem uvodu v aplikacijo ali po n-ti prijavi v aplikacijo, je uporabnik nato preusmerjen v glavni meni, (Slika 5.1). V glavnem meniju se uporabnik lahko premika levo in desno in izbira med tremi podmeniji. Podmeniji, ki so na voljo, so: "Asanas", "Goals" in "Notes". V glavnem meniju se zgoraj desno nahaja tudi gumb za odjavo.

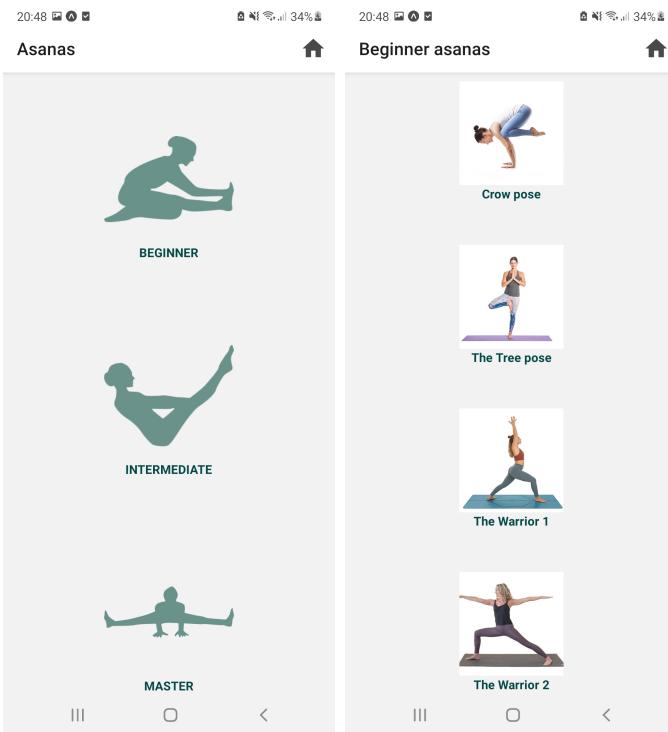
Podmeni *Asanas* je razdeljen na tri dodatne podmenije, ki določajo različne



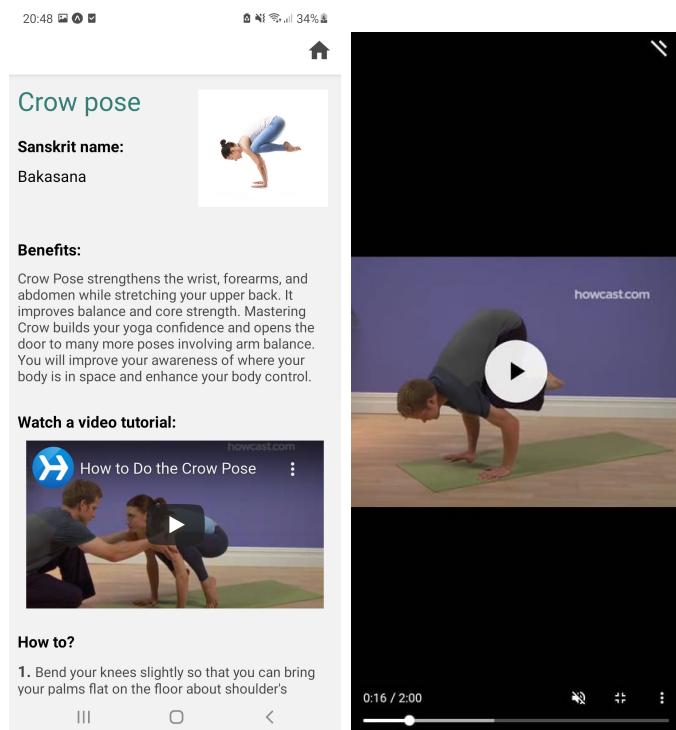
Slika 5.4: Glavni meni s tremi možnostmi izbire podmenija (*Asanas, Goals, Notes*).

kategorije težavnosti položajev. Kategorije težavnosti so: *Begginer, Intermediate, in Master*, (Slika 5.5).

S klikom na eno izmed naštetih težavnosti se uporabniku odpre dodaten podmeni, kjer se nahaja seznam z imeni in slikami položajev določene težavnosti. S klikom na enega izmed položajev, se odpre stran z angleškim imenom, sanskrit imenom, sliko, pozitivnimi učinki položaja, razdelek *How to* z opisom izvedbe po korakih in videoposnetkom položaja za ogled same izvedbe, (Slika 5.5). Uporabnik ima možnost ogleda videoposnetka položaja, s klikom na videoposnetek. Videoposnetek se predvaja neposredno v aplikaciji, kjer ima uporabnik tudi možnost povečave videa skozi celoten ekran mobilne naprave, s klikom na naslov posnetka pa si ga uporabnik lahko ogleda tudi izven aplikacije na platformi Youtube.

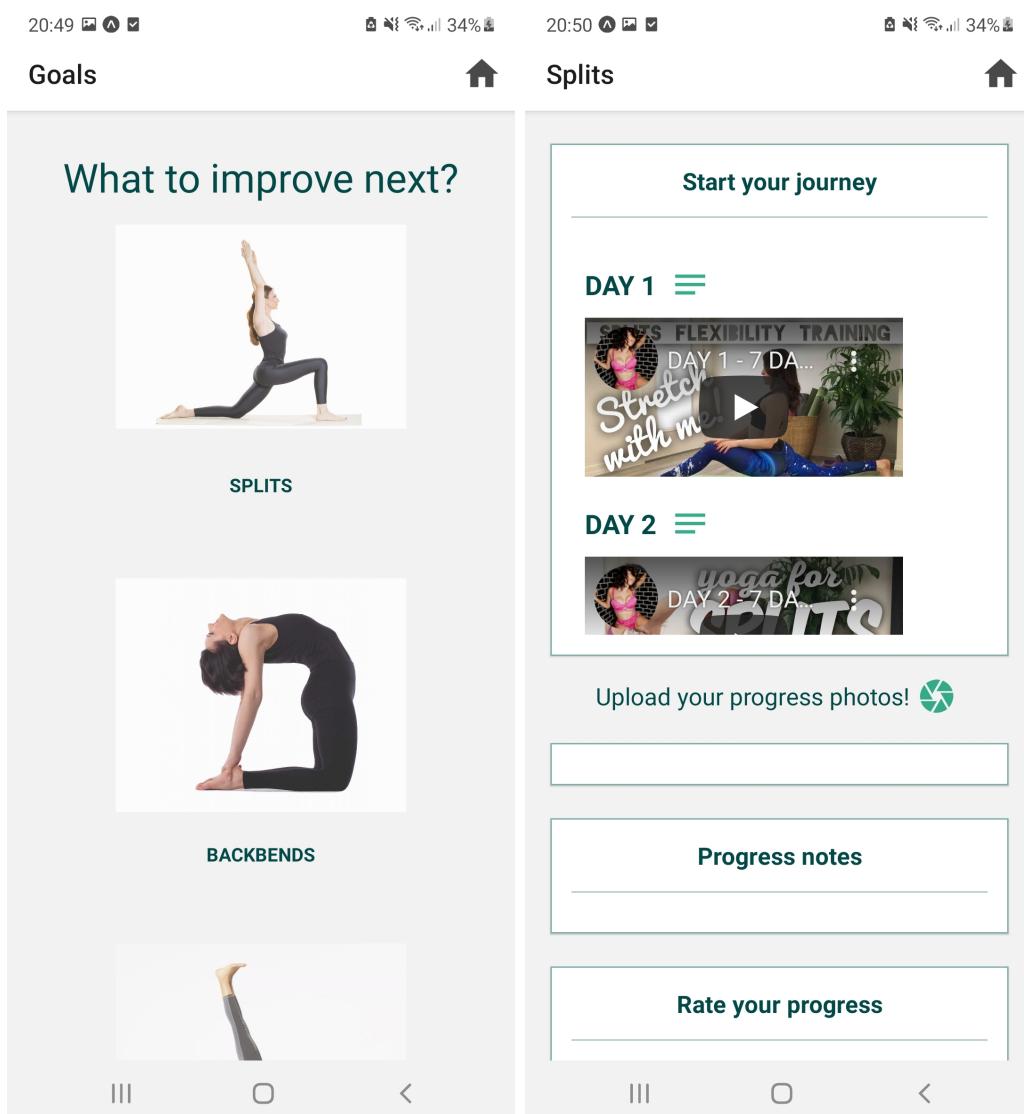


Slika 5.5: Podmeni s težavnostmi položajev (levo) in seznam položajev v podmeniju *Begginer* (desno)



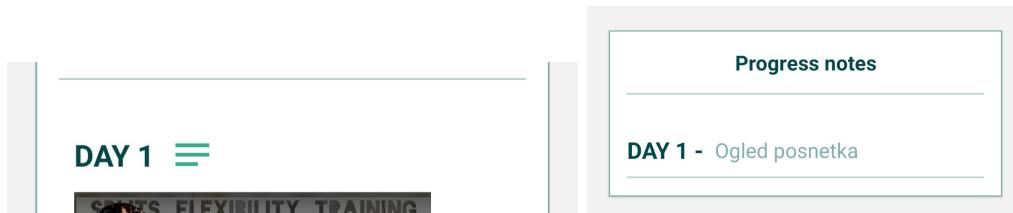
Slika 5.6: Stran s podrobnostmi položaja (desno) in predvajanje videa v aplikaciji (desno)

Podmeni *Goals* se prav tako razdeli na tri dodatne podstrani, ki predstavljajo tri kategorije, v katerih se lahko uporabnik izboljša in spremlja svoj napredok. Kategorije so: *Splits*, *Backbends* in *Inversions*. Ko si uporabnik izbere eno izmed kategorij, se odpre stran s sedem dnevnim programom različnih videoposnetkov, ki jim uporabnik lahko sledi za lasten napredok, (Slika 5.7).



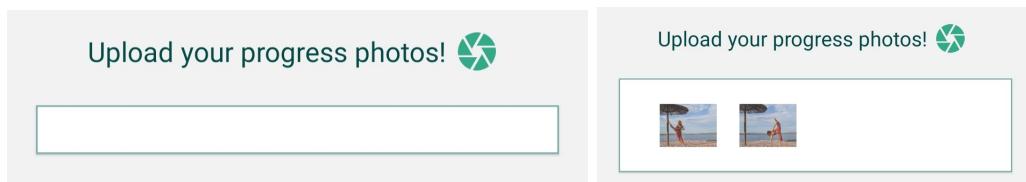
Slika 5.7: Podmeni *Goals* (levo) in podmeni *Splits* (desno).

Tako kot pri položajih, se tudi tukaj videoposnetek predvaja in odpre neposredno v aplikaciji, brez odpiranja novega okna izven aplikacije. Poleg dneva programa se nahaja ikona za dodajanje zapiska, (Slika 5.8). S klikom na ikono lahko uporabnik lahko zapisek vezan na dan v programu, zapisek pa se potem pojavi v razdelku *Progress notes*.



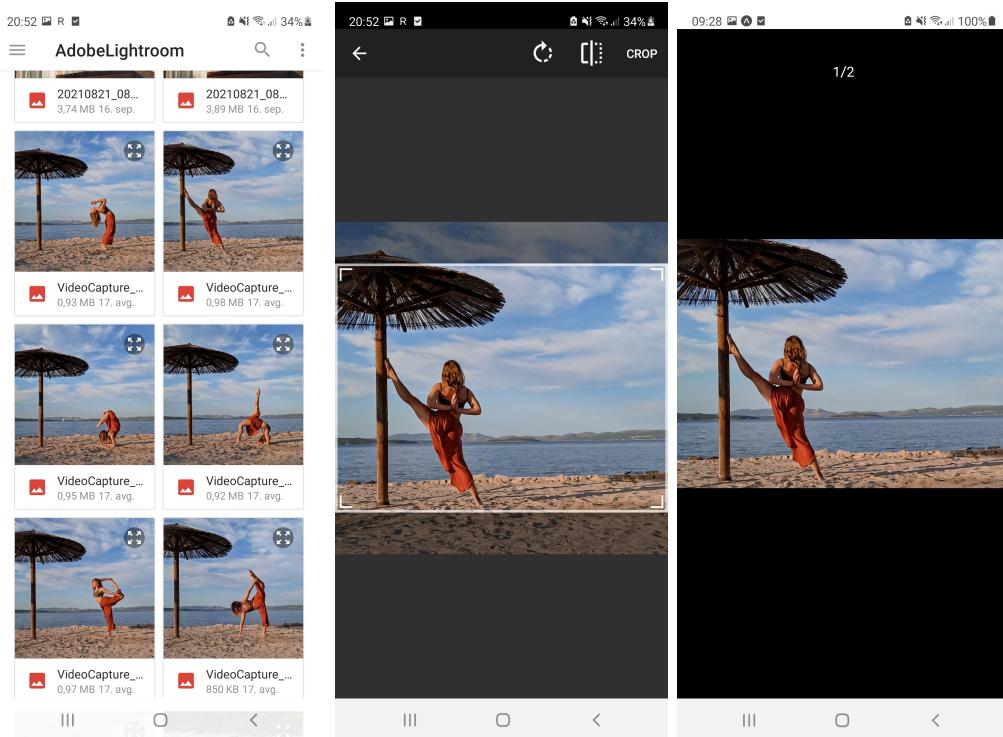
Slika 5.8: Ikona poleg dneva v programu (levo) in razdelek Progress notes v katerem se nahaja seznam zapiskov po dnevih (desno).

Pod programom se nahaja možnost dodajanja slik, s katerimi uporabnik lažje sledi napredku, (Slika 5.9). S klikom na ikono fotoaparata se odpre knjižnica s slikami in videoposnetki na mobilni napravi.



Slika 5.9: Razdelek za dodajanje slik (levo) in prikaz naloženih slik v istem razdelku (desno).

Po izboru slike se uporabniku odpre vmesnik za urejanje slike. V vmesniku uporabnik lahko sliko obrača, zrcali ali obreže, (Slika ??). Po končanem urejanju se s klikom na *CROP* slika naloži v mobilno aplikacijo, kjer je vidna v razdelku s slikami. Za ogled slik uporabnik enostavno klikne na sliko, ki se odpre v pregledovalniku slik, med slikami pa se uporabnik premika z drsanjem v levo ali desno.



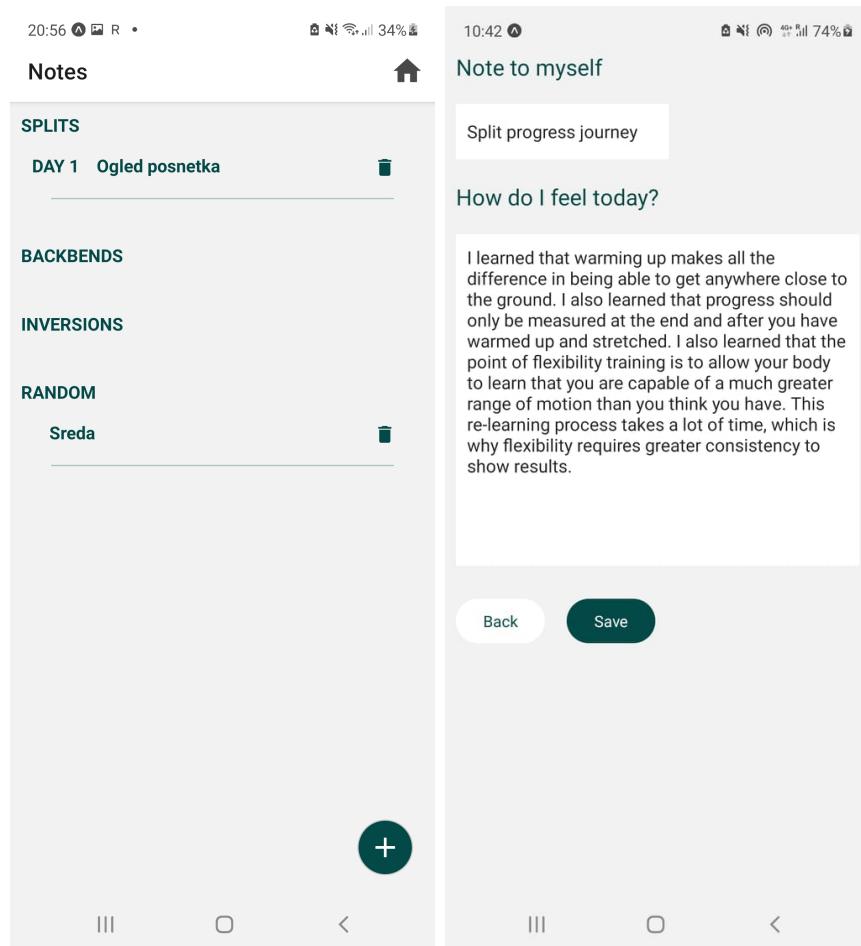
Slika 5.10: Knjižnica slik na napravi (levo), urejevalnik izbrane slike (sredina) in pregledovalnik slik (desno).

Uporabnik si lahko pred, med ali po programu poda samooceno, s katero lahko lažje spremi svoj lasten napredek in kakšen rezultat je dosegel, (Slika 5.11). S klikom na določeno število zvezdic se le-te obarvajo in izpiše se opis ocene.



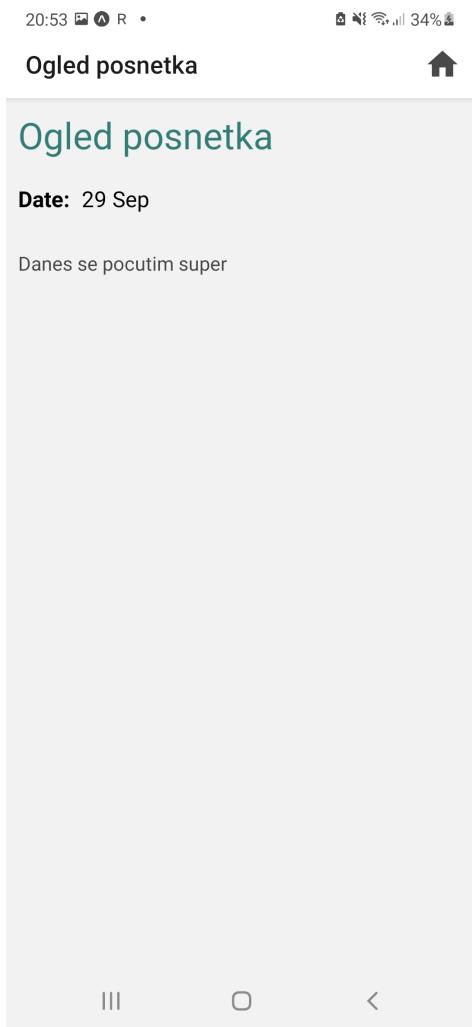
Slika 5.11: Prazen razdelek za podajanje ocene napredka (levo) in podana ocena (desno).

Podmeni *Notes* ima seznam zapiskov, ki jih je uporabnik ustvaril, ter spodaj desno gumb za dodajanje novega zapiska, (Slika 5.12). Vsak zapisek v seznamu je razvrščen v ustrezeno kategorijo in prikazan z imenom na levi strani vrstice, na desni strani vrstice pa se nahaja ikona koša za smeti, s klikom na katerega lahko uporabnik izbriše zapisek. S klikom na gumb za dodajanje zapiska, se uporabniku odpre forma za vnos naslova zapiska in okno za vnos vsebine zapiska. Pod oknom za vsebino se nahajata še dva gumba, levo gumb *Back*, ki uporabnika vrne v seznam zapiskov in desno gumb *Save*, ki omogoča shranitev zapiska.



Slika 5.12: seznam zapiskov (levo) in forma z vneseno vsebino zapiska (desno).

Po shranitvi zapiska je uporabnik samodejno preusmerjen nazaj na seznam zapiskov. S klikom na izbrani zapisek se odpre stran z naslovom zapiska, datumom, kdaj je bil zapisek ustvarjen in zapisano besedilo uporabnika, (Slika 5.13).



Slika 5.13: Prikaz strani z ogledom podrobnosti zapiska (desno).

Za odjavo iz aplikacije lahko uporabnik v vsakem od podmenijev klikne gumb *Home* (ikona hiške zgoraj desno), ki ga preusmeri v glavni meni, kjer se na istem mestu nahaja gumb za odjavo.

Poglavlje 6

Sklepne ugotovitve

V sklopu diplomske naloge smo raziskovali področje mobilnih aplikacij namenjenih učenju joge in jih primerjali z našo aplikacijo. Ugotovili smo, da imajo nak skupnih funkcionalnosti, vsaka pa ima tudi svojo dodano vrednost, ki privablja različne tipe uporabnikov. V diplomskem delu smo opisali katera orodja in tehnologije so bile uporabljene pri razvoju mobilne aplikacije, opisali smo korake načrtovanja ter predstavili mobilno aplikacijo. Na koncu pa smo opisali še možen načrt za nadaljnji razvoj.

Menimo, da smo v diplomskem delu prišli do cilja, ki smo si ga zastavili, imamo pa s trenutno verzijo aplikacije še kar nekaj možnega prostora za razvoj in nadgradnjo. Veliko časa smo namenili zbiranju in prepisovanju podatkov, saj je številni nabor asan pomemben za boljšo uporabniško izkušnjo. Prav tako smo veliko časa namenili raziskovanju in načrtovanju razvoja aplikacije, da smo izbrali najbolj primerna orodja in tehnologije za sam razvoj. Z izdelavo prototipa nismo imeli težav, saj smo na tem področju imeli že kar nekaj izkušenj. Največji izziv pri razvoju pa nam je predstavljala prijava in avtentifikacija uporabnika.

Končen rezultat diplomske naloge je mobilna aplikacija, ki uporabnikom omogoča učenje jogijskih položajev, učenje in izboljšanje osebnih ciljev, nalanjanje slik za lažje spremeljanje napredka in ustvarjanje zapiskov, kar zaokroži celoten učni proces, ki smo ga želeli ustvariti s to aplikacijo.

6.1 Ideje za nadaljnji razvoj

Prva ideja za nadaljnji razvoj je razširitev baze podatkov, s katero bi imeli uporabniki dostop do več vsebine za učenje.

Naslednja možna nadgradnja bi lahko bil bolj personaliziran program za osebni napredek uporabnika, tako da bi se posnetki in programi ustvarjali samodejno glede na podano oceno uporabnika. Na primer, v kolikor bi uporabnik podal oceno svojega napredka 6 in ocene nekaj časa ne bi spremenjal, bi mu aplikacija tekom časa ponudila nov program, kateremu bi uporabnik sledil za napredek.

Prav tako bi ena izmed možnih nadgradenj bila povezovanje uporabnikov z drugimi uporabniki, tako da bi si lahko ogledali profil drugega uporabnika in preverili njegov napredek, kar bi morda nekatere uporabnike še dodatno motiviralo.

Zadnja ideja pa je razširitev uporabniških profilov, s katerim bi že obstoječim učiteljem omogočili sestavljanje in objavljanje programov, katerim bi potem drugi uporabniki lahko sledili.

Literatura

- [1] arhitektura. Dosegljivo: http://colos.fri.uni-lj.si/ERI/RACUNALNISTVO/PODATKOVNE_BAZE/trinivojska_arhitektura.html. [Dostopano: 25. 9. 2021].
- [2] Asane. Dosegljivo: <https://www.vodenevadbe.si/post/kaj-pomeni-beseda-asana>. [Dostopano: 26. 5. 2021].
- [3] Expo. Dosegljivo: <https://docs.expo.io/>. [Dostopano: 25. 6. 2021].
- [4] Express. Dosegljivo: <https://expressjs.com/>. [Dostopano: 21. 7. 2021].
- [5] Git. Dosegljivo: <https://git-scm.com/>. [Dostopano: 15. 8. 2021].
- [6] O jogi. Dosegljivo: <https://www.hathajoga.si/joga>. [Dostopano: 20. 5. 2021].
- [7] Zgodovina joge. Dosegljivo: <https://www.hathajoga.si/zgodovina-joge>. [Dostopano: 20. 5. 2021].
- [8] Susan Hollister. *The Top 100 Best Yoga Poses Relieve Stress, Increase Flexibility, and Gain Strength*. Addison-Wesley, 2017.
- [9] Js. Dosegljivo: <https://www.theserverside.com/definition/JavaScript>. [Dostopano: 21. 7. 2021].
- [10] Mysql. Dosegljivo: <https://www.digitalocean.com/community/tutorials/what-is-mysql>. [Dostopano: 15. 8. 2021].

- [11] npm. Dosegljivo: <https://docs.npmjs.com/cli/v6/commands/npm>. [Dostopano: 25. 6. 2021].
- [12] Nataša Petavs. *Priročnik za usposabljanje za naziv učitelja joge*. 2021.
- [13] Rn. Dosegljivo: <https://www.netguru.com/glossary/react-native>. [Dostopano: 25. 6. 2021].
- [14] Sql. Dosegljivo: http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PODAT_PLAT_RAZ_PO/sql.html. [Dostopano: 25. 8. 2021].
- [15] Tableplus. Dosegljivo: <https://mariadb.com/kb/en/tableplus/>. [Dostopano: 20. 7. 2021].
- [16] trackyoga. Dosegljivo: <https://trackyoga.app/>. [Dostopano: 2. 9. 2021].
- [17] Ts. Dosegljivo: <https://www.typescriptlang.org/>. [Dostopano: 20. 7. 2021].
- [18] Vsc. Dosegljivo: <https://www.educative.io/edpresso/what-is-visual-studio-code>. [Dostopano: 15. 7. 2021].
- [19] verywellfit. Dosegljivo: <https://www.verywellfit.com>. [Dostopano: 5. 6. 2021].
- [20] Wireframepro. Dosegljivo: <https://www.mockflow.com/apps/wireframepro/>. [Dostopano: 25. 9. 2021].
- [21] yoga108. Dosegljivo: https://play.google.com/store/apps/details?id=com.yogiclife&hl=en_US&gl=US. [Dostopano: 2. 9. 2021].