

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Klementina Garbajs

# **Mobilna aplikacija Asana Master**

DIPLOMSKO DELO

UNIVERZITETNI 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  $\text{\LaTeX}$ .*

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Besedilo teme diplomskega dela študent prepíše iz študijskega informacijskega sistema, kamor ga je vnesel mentor. V nekaj stavkih bo opisal, kaj pričakuje od kandidatovega diplomskega dela. Kaj so cilji, kakšne metode uporabiti, morda bo zapisal tudi ključno literaturo.



*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

<b>1</b>	<b>Uvod</b>	<b>1</b>
1.1	Pregled sorodnih aplikacij . . . . .	2
1.2	Struktura diplome . . . . .	3
<b>2</b>	<b>Joga</b>	<b>5</b>
2.1	O jogi . . . . .	5
2.2	Zgodovina joge . . . . .	6
2.3	Asane in njihov pomen . . . . .	8
<b>3</b>	<b>Opis aplikacije in načrtovanje</b>	<b>11</b>
3.1	Ideja in cilji aplikacije . . . . .	11
3.2	Funkcionalnosti aplikacije . . . . .	12
3.3	Načrtovanje aplikacije in zbiranje podatkov . . . . .	12
<b>4</b>	<b>Uporabljena tehnologija in orodja</b>	<b>17</b>
4.1	npm . . . . .	17
4.2	React Native . . . . .	18
4.3	Expo . . . . .	18
4.4	Express . . . . .	19
4.5	JavaScript . . . . .	19
4.6	TypeScript . . . . .	19

4.7	Visual Studio Code . . . . .	20
4.8	MySQL . . . . .	20
4.9	TablePlus . . . . .	21
4.10	Git . . . . .	21
<b>5</b>	<b>Predstavitev aplikacije</b>	<b>23</b>
5.1	Glavne funkcionalnosti in delovanje aplikacije . . . . .	23
<b>6</b>	<b>Sklepne ugotovitve</b>	<b>31</b>
6.1	Ideje za nadaljni razvoj . . . . .	32
	<b>Literatura</b>	<b>33</b>



# Seznam uporabljenih kratic

kratica	angleško	slovensko
IDE	integrated development enviroment	integrirano razvojno okolje



# Povzetek

**Naslov:** Mobilna aplikacija Asana Master

**Avtor:** Klementina Garbajs

Za diplomsko delo smo se odločili razviti mobilno aplikacijo Asana Master, ki bi v neki celoti zaobjela potovanje posameznika po njegovi poti Joge. 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, spremljanjem napredka in sledenja lastnim občutkom praktikantov, smo razvili aplikacijo, 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.

**Ključne besede:** Joga, Mobilna aplikacija.



# Abstract

**Title:** Mobile app Asana Master

**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. Asane oz. 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.



# Poglavje 1

## Uvod

Joga je za nekoga lahko le oblika sprostitve, način napajanja z energijo, vadba za boljše počutje, za tiste, ki se z jogo ukvarjajo bolj intenzivno, pa je joga 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 seveda 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 joga položajev in pravilne izvedbe le-teh zelo pomembno. In kaj je asana? Sanskrtski izraz asana pomeni poza ali položaj telesa. Asana je psiho-somatska jogiska 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 spremljanjem napredka pri tem, smo se za diplomsko nalogo odločili razviti mobilno aplikacijo, ki bi bodočim joga učiteljem in tudi vsem ostalim praktikantom joge omogočala lažje in bolj zabavno obliko učenja, ter dodatno motivacijo za napredek. Aplikacijo smo razvili z uporabo ogrodja React Native in odprtokodno platformo Expo. Za pisanje kode smo uporabili jezike, Java Script,

TypeScript in SQL. Aplikacija teče na Expotovem lokalnem strežniku, za povezavo do baze podatkov pa smo uporabili strežnik Express. Podatki so shranjeni v MySql bazi, ki teče preko MySQL strežnika.

## 1.1 Pregled sorodnih aplikacij

Pri pregledu sorodnih aplikacij smo prišli do ugotovitev, da slovenske aplikacije na trgu ni, saj je slovenski trg žal premajhen.

Ena izmed boljših aplikacij, ki jo trenutno uporabljamo tudi sami, je aplikacija Track Yoga, 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 itd,... V aplikaciji ima vsak uporabnik tudi svoj profil, kjer si lahko nastavi tedenske cilje, spremlja koliko programov in vodenih vadb je že opravil, sledi svojemu napredku z zbiranjem značk (angl. Badge) itd.

Aplikacija je torej na nek način podobna naši, vendar pa nima možnosti delanja zapiskov in ne omogoča spremljanja lastnega napredka, kot je objavljane slik, podajanje samoocene in interaktivno sledenje programu, kar uporabnika še bolj motivira.

Ena izmed aplikacij, ki so osredotočene bolj na učenje posameznika o asanah, pa je aplikacija Yoga 108. 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. Prav tako, kot že prej omenjena aplikacija, tudi ta ne omogoča delanja zapiskov in spremljanja napredka.

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.



## 1.2 Struktura diplome

Diplomsko delo smo razdelili na šest večjih poglavij. Poglavje *Uvod* smo namenili predstavitvi sorodnih del, ter splošni strukturi diplomske naloge, zato je tudi razdeljeno na več podpoglavij. Sledi poglavje *Joga* v katerem smo predstavili pojem Joga, zgodovino joge in pomen asan. Poglavje *Opis aplikacije in načrtovanje* smo namenili predvsem predstavitvi glavnih funkcionalnosti, ki jih aplikacija ponujam, ter arhitekturi sistema. V naslednjem poglavju *Uporabljena tehnologija in orodja* smo predstavili uporabljene tehnologije ter orodja, za razvoj aplikacije. V poglavju *Predstavitev aplikacije* smo s pomočjo zaslonskih posnetkov prikazali glavne funkcionalnosti aplikacije, zaključno poglavje pa smo namenili sklepnim ugotovitvam in idejam za možni nadaljnji razvoj.



## Poglavje 2

### Joga

#### 2.1 O jogi

Joga združuje načine zavedanja, zaznavanja, doživljanja, soočanja, spoznavanja in delovanja, preko katerih je človek zmožen bivati neodvisno od lastne pogojenosti. V jogijskih spisih je ta zmožnost bivanja opisana kot stanje odrešitve, saj je preko nje človek odrešen trpljenja kateremu je podvržen kot človeško bitje bivajoče v tem svetu. Joga namreč izhaja iz spoznanja, da trpljenje ni neločljivo povezano s človeškim bitjem oziroma ni neizogibno vsebovano v svetu v katerem živi. Trpljenje je šele učinek nezmožnosti delovati na določen način.

Beseda joga izhaja iz korena besede yuj, kar pomeni povezovati. Joga povezuje snovnost in duhovnost, telo in duha. Medsebojno sodelovanje telesa in duha povezuje v usklajeno, dopolnjujoče se gibanje. Če želi človek povezati svoje telo in duha na omenjeni način, se mora biti najprej zmožen razvezati oziroma oddaljiti od lastnih določujočih pogojev. [5]

## 2.2 Zgodovina joge

Kot posebna metoda preseganja človeške pogojenosti, je omenjena že v starodavnih vedskih spisih katerih nastanke zgodovinarji umeščajo na območje današnja Indije. Preden se je namreč razširila kot uveljavljena praksa so kot prevladujoči načini samo-preseganja služili predvsem obredni rituali žrtvovanja. Preko ritualov žrtvovanja, ki so vsebovali elemente smrti in ponovnega rojstva, so si ljudje poskušali zagotoviti moč nadzora nad lastno minljivostjo.

V obdobju prvih upanišadskih spisov so obredni rituali začeli izgubljati svoj vlogo prevladujočega načina samo-preseganja. Zmožnost odrešitve je prenehala obstajati kot poseben privilegij pogojen s poznavanjem ritualnih praks, saj se je iz rituala, kot sredstva odrešitve, pozornost preusmerila na osebo in način njenega bivanja. Modreci ršiji - avtorji upanišadskih spisov - so namreč v svojih nazorih poudarjali, da je odrešitev zmožen doseči prav vsak z načinom svojega bivanja in delovanja.

To bivanje in delovanje so opisali v filozofskih nazorih samkhye iz katerih izhaja tudi filozofija joge. Filozofija samkhye temelji na predpostavki, da je človeško trpljenje posledica nepoznavanja samega sebe in vzrokov svojih dejanj. Sprememba načina bivanja, ki bi onemogočila trpljenje je pogojena s spoznanjem in udejanjanjem samega sebe. To spoznanje ni razumsko spoznanje, saj spoznavajoči ne ustvarja spoznanja, ki bi bilo ločeno od njega, ampak je spoznanje oziroma se (sam sebi) razodeva kot spoznanje.

Spoznanje samega sebe se razodeva na nivoju skupne zavesti. Tu se jaz zaveda samega sebe neodvisno od fizičnega in psihičnega doživljanja. S tem, ko je zmožen kot pričujoča zavest bivati onstran fizičnega in psihičnega izkustva, lahko spoznava in spreminja pogoje lastnega bivanja in vzroke svojega delovanja.

Filozofija joge je prevzela osnovne predpostavke samkhye, 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 predpostavljajo, da vsebuje tako fizično kot psihično re-

alnost) 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 (njegovo življenje zgodovinarji umeščajo v obdobje od 2. stol. pr. n. št - 5. st. n. št.). V njih izpostavlja, kako praksa joge ne omogoča zgolj duhovne neodvisnosti od pogojev materialnega sveta, ampak tudi zmožnost njihovega spreminjanja. S pomočjo prakse joge, ki vključuje tako fizično, psihično kot duhovno komponento, smo zmožni spreminjati svojo nezavedno pogojenost in iz nje izhajajočo zmožnost delovanja. [6]

Tradicionalno jogo sestavlja osem temeljnih vej (ash – osem, anga – veja), ki jih je prvič sistematično opisal Patanjali v svojih Yoga-sutrah:

### 1. Yama ali etične zmožnosti

- 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 (brahmacarya)
- onemogočenje pohlepa (aparigraha) oziroma težnje po prilaščanju tega česar ne potrebujemo

### 2. Niyama ali discipline

- čiščenje telesa in duha (sauca)
- zmožnost izkusiti zadovoljstvo neodvisno od zunanjih dejavnikov (santosa)
- askeza (tapas)
- odsotnost besed (kastha mauna)

- odsotnost gest in kretenj (akara mauna)
- študij načinov samo-preseganja

3. **Asana ali telesni položaji**

4. **Pranayama ali dihalne tehnike**

5. **Pratyahara ali neodvisnost čutil od zaznavanja zunanjih dejavnikov**

6. **Dharana ali zbranost in zmožnost nadzorovanega usmerjanja pozornosti**

7. **Dhyana ali meditacija**

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.

V hatha jogi so asane ena osnovnih tehnik telesnega gibanja. Asane dajejo telesu gibljivost in pravilno držo, ki omogoča optimalno energetsko in telesno stabilnost, usmerjene so v živčni sistem, predvsem v hrbtenico, ki je po jogijski anatomiji osnovna energetska os našega bivanja.

Začetniki v hatha jogi so se morali najprej naučiti, kako pravilno stati, sedeti in ležati. Zato se jogijska praksa tudi bistveno razlikuje od vseh drugih vadb, saj zahteva poleg natančnega, pravilnega izvajanja asan tudi pozornost, usmerjenost uma v to, kar počnemo. V asani smo, kot pove že sanskrito ime "as" oz. biti, tukaj in zdaj.

Asan je nešteto, nekateri viri navajajo številko 70.–80.000. Poimenovali so jih po živalih, rastlinah, mineralih in božanstvih. Bilo naj bi jih toliko, kolikor je oblik življenja na zemlji. 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 dožemanju 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. [1] Bistvo izvajanja asane ni nujno v popolni izvedbi. Bistvo asane je v popolnem medsebojnem zavedanju duše in telesa.





## Poglavje 3

# Opis aplikacije in načrtovanje

### 3.1 Ideja in cilji aplikacije

Živimo v svetu, kjer je imeti veliko mobilnih aplikacij nekaj običajnega, saj za vsako aktivnost, ki si jo izmislimo, obstaja mobilna aplikacija. A ko se želiš nekaj naučiti oz. v nečem izboljšati, ti lahko ravno prevelik nabor različnih aplikacij predstavlja oviro. V kolikor morda nisi najbolj organiziran človek, ti lahko možnost izbire ustvari še večjo zmedo. Prav iz tega razloga, se nam je porodila ideja o aplikaciji Asana Master, kjer ima uporabnik zgolj z eno aplikacijo možnost učenja o različnih položajih v jogi, možnost spremljanja 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. Torej vse zapakirano na enem mestu.

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, zato se bomo vedno bolj nagibali k aplikacijam, ki omogočajo več večjih funkcionalnosti v enem.

## 3.2 Funkcionalnosti aplikacije

Glavne funkcionalnosti, ki jih ima uporabnik na voljo z aplikacijo AsanaMaster so sledeče:

- **Registracija:** kreiranje novega uporabniškega računa
- **Prijava:** prijava v mobilno aplikacijo z uporabniškim računom
- **Pozabljeno geslo:** resetiranje pozabljenega gesla s pošiljanjem novega gesla na elektronski naslov uporabnika
- **Ogled videoposnetkov:** ogled videoposnetkov o asanah in programa v mobilni aplikaciji
- **Označevanje asane:** označevanje naučene asane
- **Ocenjevanje napredka:** ocenjevanje lastnega napredka uporabnika
- **Nalaganje slik:** nalaganje slik napredka uporabnika
- **Ogled slik:** ogled slik napredka uporabnika
- **Kreiranje novega zapiska:** uporabnik lahko kreira nov zapis
- **Brisanje zapiska:** uporabnik lahko željeni zapis izbriše

## 3.3 Načrtovanje aplikacije in zbiranje podatkov

Arhitektura aplikacije je sestavljena iz treh glavnih nivojev: podatkovne baze, REST APIja in mobilne aplikacije.

### 3.3.1 Podatkovna baza

Podatkovna baza je zbirka povezanih podatkov. Prednosti uporabe podatkovne baze so možnost shranjevanja večjih količin podatkov, hitrejši prenos podatkov in hitrejša obdelava podatkov. Za podatkovno bazo smo uporabili MySQL Server, za upravljanje le te pa program TablePlus. Naša podatkovna baza je sestavljena iz šestih tabel: *Users*, *Asanas*, *Goals*, *Images*, *Ratings* *Notes* in je povezana z APIjem, ki nad njimi izvaja različne operacije.

### 3.3.2 REST API

Za razvoj REST APIja smo se odločili uporabiti ogrodje Express.

```
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 3.1: Koda za vzpostavitev REST API serverja.

API je v naši aplikaciji posrednik med podatkovno bazo in uporabniki aplikacije. Sprejema zahteve s strani uporabe aplikacije in izvaja ustrezne operacije na podatkovni bazi. Uporabili smo naslednje zahteve HTTP za različne namene:

- **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

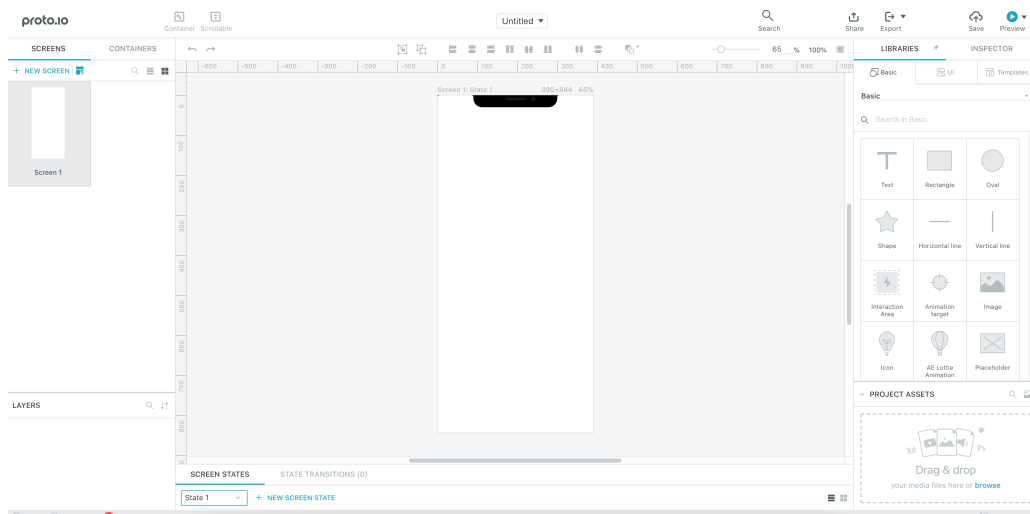
### 3.3.3 Mobilna aplikacija

Za razvoj mobilne aplikacije smo uporabili 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.

Načrtovanje same mobilne aplikacije smo začeli z izdelavo prototipa uporabniškega vmesnika mobilne aplikacije, saj je izdelava dobrega uporabniškega vmesnika pomemben sestavni del oblikovanja uporabniške izkušnje. Pri oblikovanju mobilne aplikacije se je pomembno usmeriti tako na estetiko kot seveda, tudi na funkcionalnost aplikacije. Vmesnik smo poskusili narediti čim bolj pregleden in enostaven za uporabo. Prav s tem namenom so v aplikacijo dodane tudi slike in videoposnetki.

Prototip smo izdelali s pomočjo spletne platforme za izdelavo prototipov Proto.io, (Slika 3.2). Izdelani so bili nekateri ključni zasloni aplikacije, glavne funkcionalnosti aplikacije, povezave med njimi in oblikovano glavno ozadje aplikacije na začetnem prijavnem zaslonu, ter ostali gradniki aplikacije.

Izdelavi prototipa je sledila raziskava tehnologij in primernih orodij za razvoj mobilne aplikacije, pri čemer smo primerjali predvsem različna orodja za izdelavo mobilne aplikacije. Primerjali smo 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, smo prišli do



Slika 3.2: Spletna platforma za ustvarjanje prototipov, Protoio.

zaključka, da bomo za razvoj aplikacije uporabili Expo.

Po začetnem načrtovanju izdelave smo se nato lotili še zbiranja podatkov o jogi in položajih v jogi. Podatki o jogi in položajih 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 [7]. Vir nekaterih slik je tudi zgoraj omenjena knjiga, nekatere pa smo pridobili preko spletne strani Verywellfit [15]. Videoposnetke, ki uporabnikom omogočajo učenje in izboljšanje položajev ter napredovanje, pa smo poiskali na platformi, za deljenje videoposnetkov, Youtube, (Slika 3.3).

```
CREATE TABLE `asanas` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `name` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL,  
  `description` varchar(500) COLLATE utf8mb4_unicode_ci NOT NULL,  
  `image` varchar(300) COLLATE utf8mb4_unicode_ci NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `name` (`name`)  
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

Slika 3.3: Primer poizvedbe za kreiranje tabele asanas.



## Poglavje 4

# Uporabljena tehnologija in orodja

Mobilna aplikacijo smo razvili z uporabo ogrodja React Native in ogrodja Expo, kar omogoča izdelavo mobilnih aplikacij tako za operacijski sistem Android kot iOS. Za pisanje kode smo uporabili jezika TypeScript in JavaScript. Kodo aplikacije smo pisali v programu Visual Studio Code. Za shranjevanje podatkov, ki so prikazani v aplikaciji, smo uporabili relacijsko podatkovno bazo MySQL, ki je povezana z lokalnim strežnikom. Za testiranje aplikacije smo uporabili orodje React DevTools, (Slika 4.1), ki omogoča enostavno testiranje aplikacij, tako na spletnem vmesniku, mobilnem simulatorju, kot tudi na mobilni napravi. Vse spremembe v kodi, ter zgodovina spremenjenih verzij aplikacije so shranjene v programu Git.

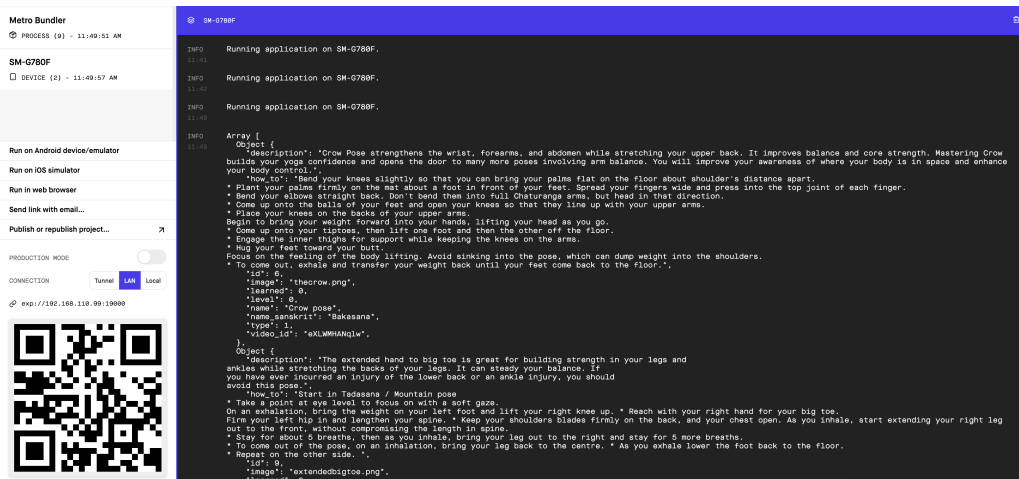
### 4.1 npm

Npm je upravitelj paketov za platformo Node JavaScript. Namešča module, tako da jih vozlišče lahko najde, in inteligentno obvladuje konflikte med odvisnostmi. Izjemno prilagodljiv je za podporo najrazličnejšim primerom uporabe. Najpogosteje se uporablja za objavljjanje, odkrivanje, nameščanje in razvoj vozliščnih programov. [10]

## 4.2 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.

React Native, je Facebook prvič izdal kot odprtokodni 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. [11]



Slika 4.1: React DevTools testno okolje na spletnem vmesniku.

## 4.3 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. 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 nastavitve IDE-jev, specifičnih za platforme, kot je to v primeru React Native CLI, in ke zato celoten postopek namestitve veliko bolj enostaven. [2]

## 4.4 Express

Express je minimalen in prilagodljiv okvir spletnih aplikacij Node.js, ki ponuja robusten nabor funkcij tako za spletne, kot mobilne aplikacije. Z neštetiimi metodami pripomočkov HTTP in vmesno programsko opremo, ki jih imamo na voljo, je omogočeno enostavno in hitro ustvarjanje robustnega API -ja. [3]

## 4.5 JavaScript

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 programskim jezikom Java, s katerim imata kar nekaj skupnih lastnosti. Na splošno velja, da so skriptni jeziki enostavnejši in hitrejši v primerjavi s kodo v bolj kompleksnih programskih jezikih kot C in C++. Pri skriptnih jezikih v splošnem obdelava kode traja dalje kot pri zbirnih jezikih, vendar pa so veliko bolj uporabni pri krajših programih zaradi njihove enostavnosti. Uporabnost JavaScript-a pride najbolj do izraza pri razvoju dinamičnih spletnih strani in dodajanju interaktivnosti na zeleno stran. [8]

## 4.6 TypeScript

TypeScript je odprtokodni jezik, ki temelji na JavaScriptu, enem izmed najbolj uporabljenih jezikov na svetu, z dodajanjem statičnih definicij tipa. 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 kjerkoli, kjer deluje JavaScript: v brskalniku, na Node.JS ali v aplikacijah. [13]

## 4.7 Visual Studio Code

Za pisanje kode smo uporabili 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. Čeprav je urejevalnik razmeroma lahek, vključuje nekaj zmogljivih funkcij, zaradi katerih je VS Code v zadnjem času postalo eno 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. [14]

## 4.8 MySQL

Prilagodljiv in zmogljiv program, MySQL je najbolj priljubljen odprtokodni sistem zbirk podatkov na svetu. Kot del široko uporabljenega svežnja tehnologij LAMP (ki ga sestavljajo 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, spletna mesta in storitve. [9]

## 4.9 TablePlus

Za pregledovanje in urejanje baze podatkov je bil uporabljen program TablePlus. 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. [12]

## 4.10 Git

Git je brezplačen in odprtokodni porazdeljeni sistem za nadzor različic, zasnovan za hitro in učinkovito obdelavo vsega, od majhnih do zelo velikih projektov. [4]

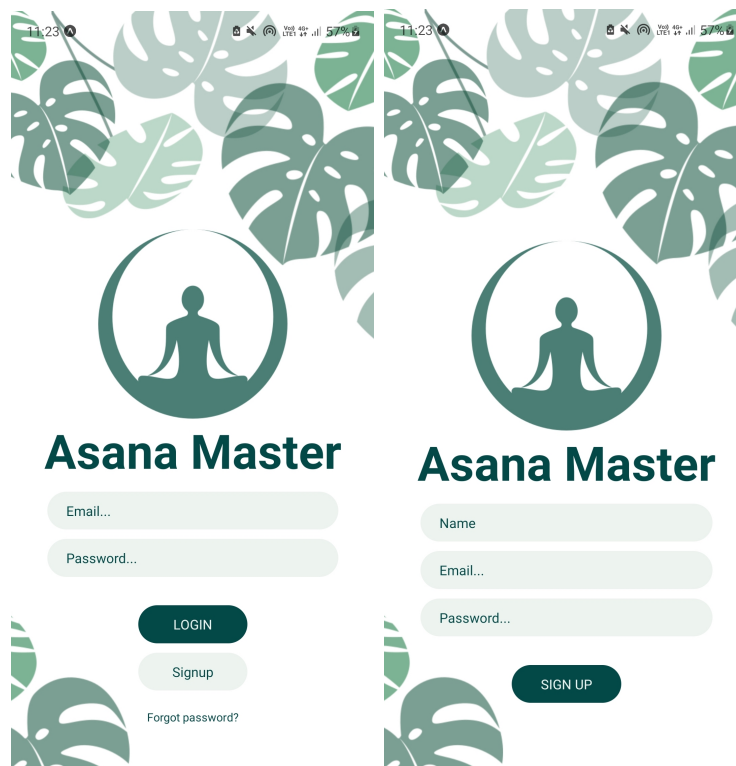


## Poglavje 5

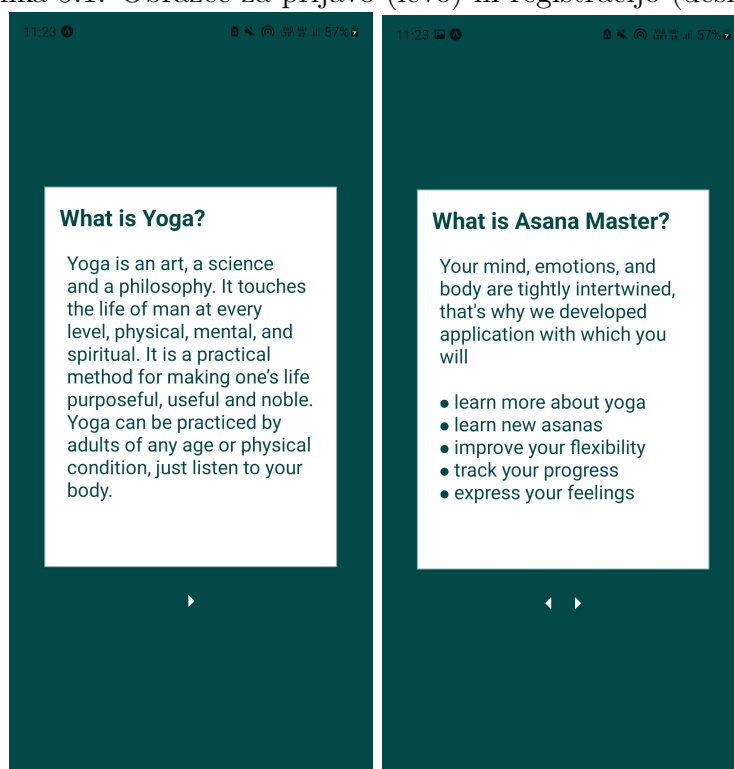
# Predstavitev aplikacije

### 5.1 Glavne funkcionalnosti in delovanje aplikacije

Ob zagonu mobilne aplikacije je uporabnik najprej preusmerjen na stran za prijavo, (Slika 5.1). 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 in, kar ga preusmeri na stran za registracijo. 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. V kolikor je uporabnik že registriran, a je pozabil svoje geslo, si lahko preko klika na gumb Forgot password?, na svoj elektronski naslov, pošlje povezavo za ponastavitev gesla. Po uspešni prijavi preko enega izmed zgoraj omenjenih načinov je uporabnik preusmerjen v jedro aplikacije. Ob prvi prijavi, uporabnika ob vstopu v aplikacijo pozdravi kratek uvod v aplikacijo, kjer se seznani s pojmom joga in s kratkim opisom bistva aplikacije. Po zaključenem uvodu v aplikacijo je uporabnik nato preusmerjen v glavni meni, ki je razdeljen na 3 glavne podmenije: Asanas, Goals in Notes.

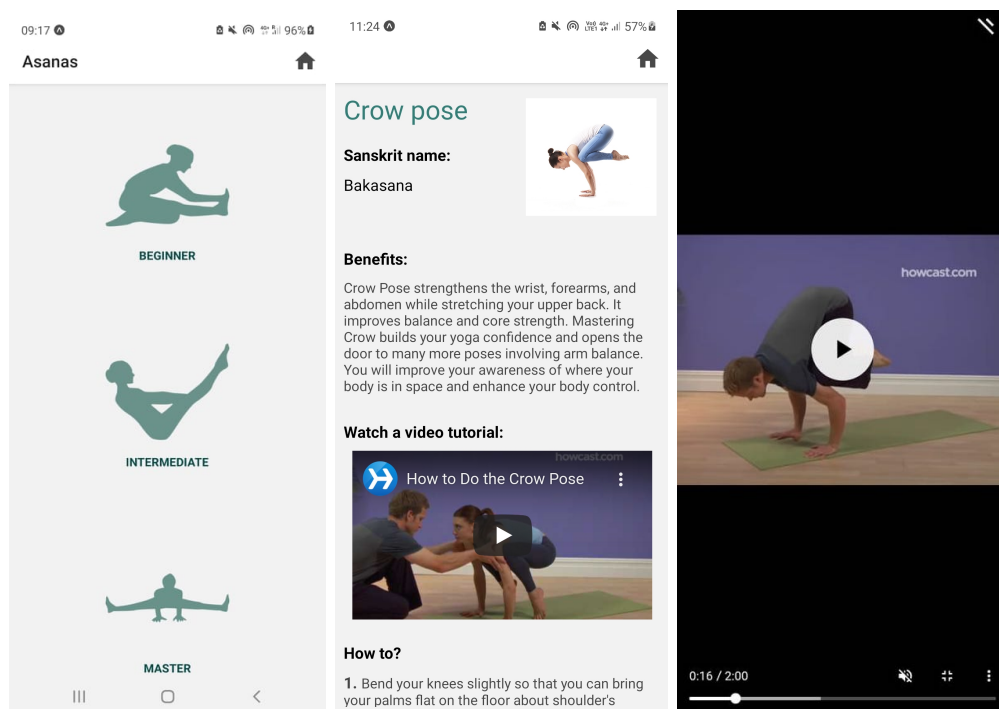


Slika 5.1: Obrazec za prijavo (levo) in registracijo (desno)



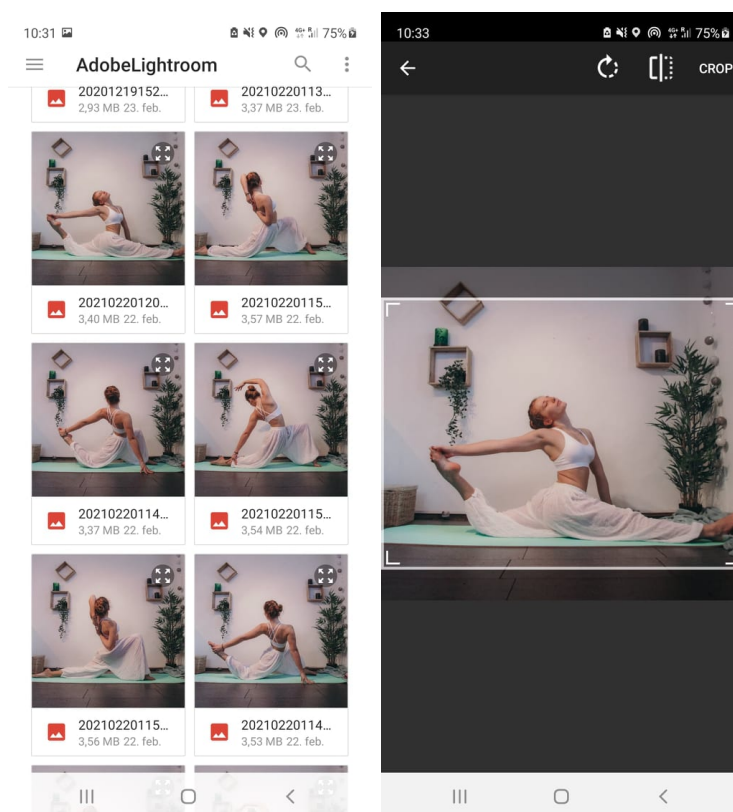
Slika 5.2: Prva in druga stran začetnega pozdrava uporabnika, ob prvi prijavi v aplikacijo.

Podmeni Asanas je razdeljen na tri dodatne podstrani, ki določajo različne težavnosti položajev, to so Begginer, Intermediate in Master. V bazi podatkov imamo pri vsakem položaju shranjeno še njegovo stopnjo težavnosti, kar nam omogoča enostavno sortiranje v omenjene težavnosti. S klikom na eno izmed težavnosti se nam odpre seznam z imeni in slikami položajev. S klikom na določen položaj, se nam odpre stran z angleškim imenom, sanskrit imenom, sliko, opisom izvedbe in videoposnetkom položaja, (Slika 5.3). Uporabnik ima možnost ogleda videopostneka položaja, s klikom na videoposnetek, (Slika 5.3). Vir videoposnetka je portal Youtube, posnetek pa se predvaja neposredno v sami aplikaciji, brez dodatnega odpiranja videoposnetka izven mobilne aplikacije. Za to funkcionalnost smo uporabili React Native knjižnico react-native-youtube-iframe.



Slika 5.3: Stran s prikazom podatkov o asani (levo) in ogled videoposnetka (desno)

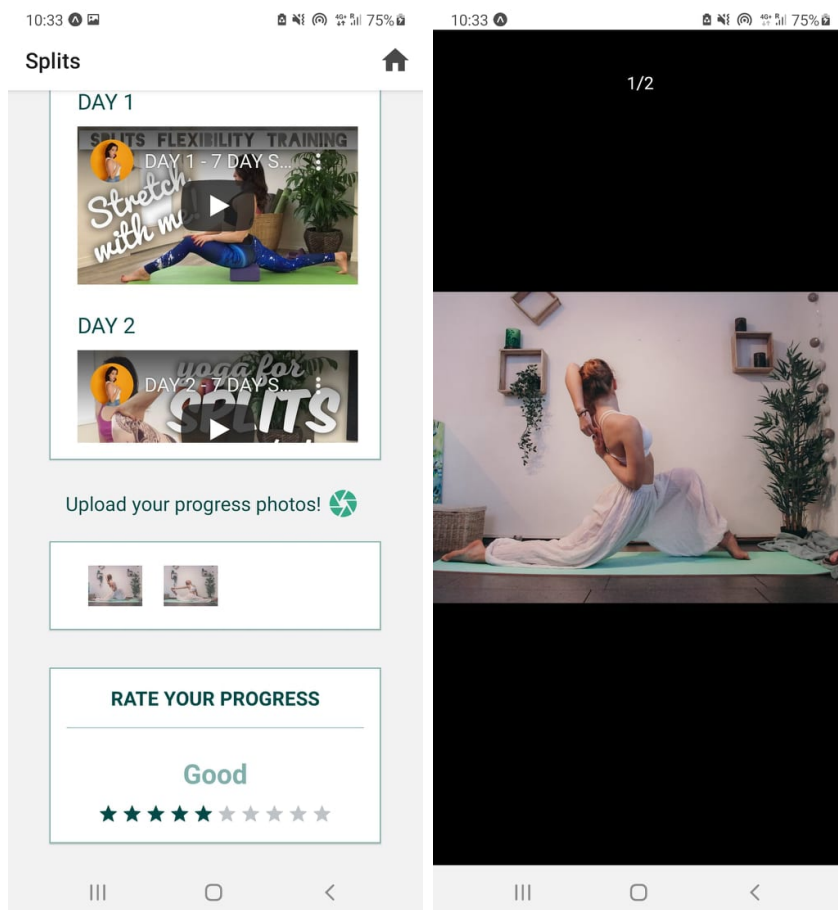
Podmeni Goals se prav tako razdeli na tri dodatne podstrani, ki predstavljajo tri kategorije, v katerih se lahko uporabnik izboljša in spremlja svoj napredek. Tri kategorije so: Splits, Backbends in Inversions. Ko si uporabnik izbere eno izmed kategorij, se odpre starn s sedem dnevnim programom različnih videoposnetkov, ki jim uporabnik lahko sledi za lasten napredek. Prav tako kot pri položajih, smo tudi tukaj naredili tako, da uporabnik predvaja in odpre videoposnetek neposredno v aplikaciji, brez odpiranja novega okna izven aplikacije. Pod programom sledi še možnost dodajanja slik, s katerimi uporabnik lažje sledi napredku. Za nalaganje slik smo uporabili Expotovo expo-image-picker, ki omogoča dostop do uporabniškega vmesnika sistema za izbiranje slik in videoposnetkov iz knjižnice telefona ali fotografiranje s fotoaparatom, (Slika 5.4).



Slika 5.4: Prikaz menija slik na mobilni napravizi izbor slike (levo) in okno za urejanje izbrane slike (desno)



Za ogled naloženih slik pa smo uporabili React Native knjižnico `react-native-image-zoom-viewer`, ki omogoča jasen in enostaven pregled slik, z možnostjo povečave, (Slika 5.5).

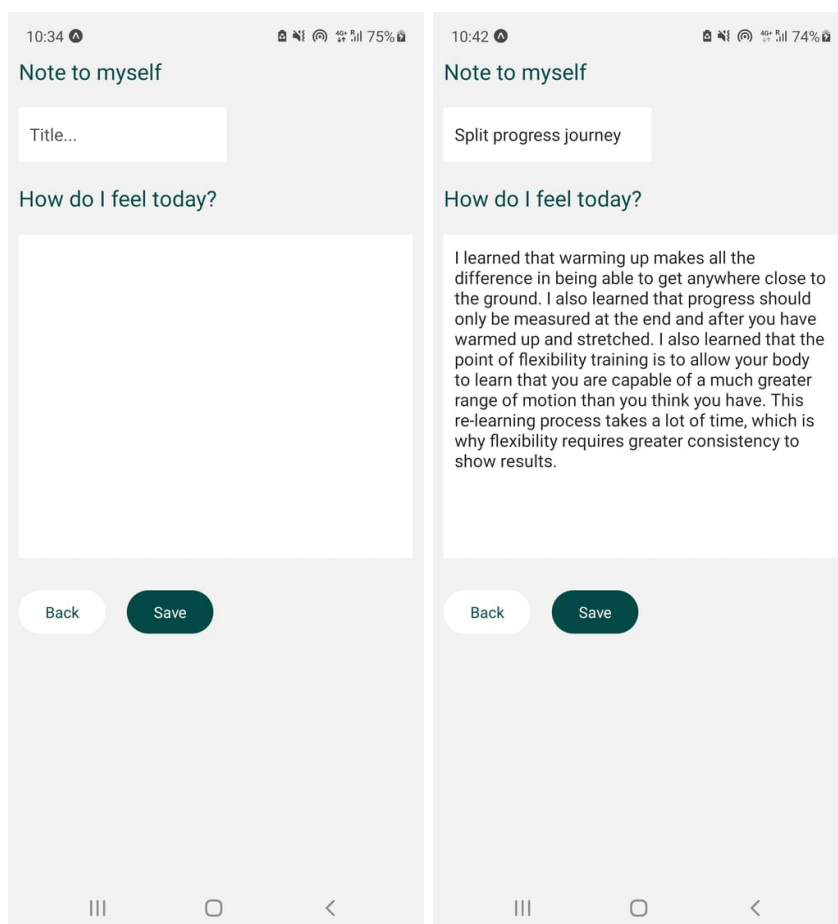


Slika 5.5: Stran splits v podmeniju Goals (levo) in okno za ogled naložene slike (desno)

Uporabnik ima poleg zgoraj naštetih funkcionalnosti še možnost podajanja samoocene, da lahko lažje spremlja svoj lasten napredek in kje na poti do cilja se nahaja.

Podmeni Notes ima seznam zapiskov, ki jih je uporabnik ustvaril, ter spodaj desno gumb za dodajanje novega zapiska. 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 2 gumba, levo gumb Back, ki uporabnika vrne v seznam zapiskov in desno gumb Save, ki omogoča shranitev zapiska, (Slika 5.6).

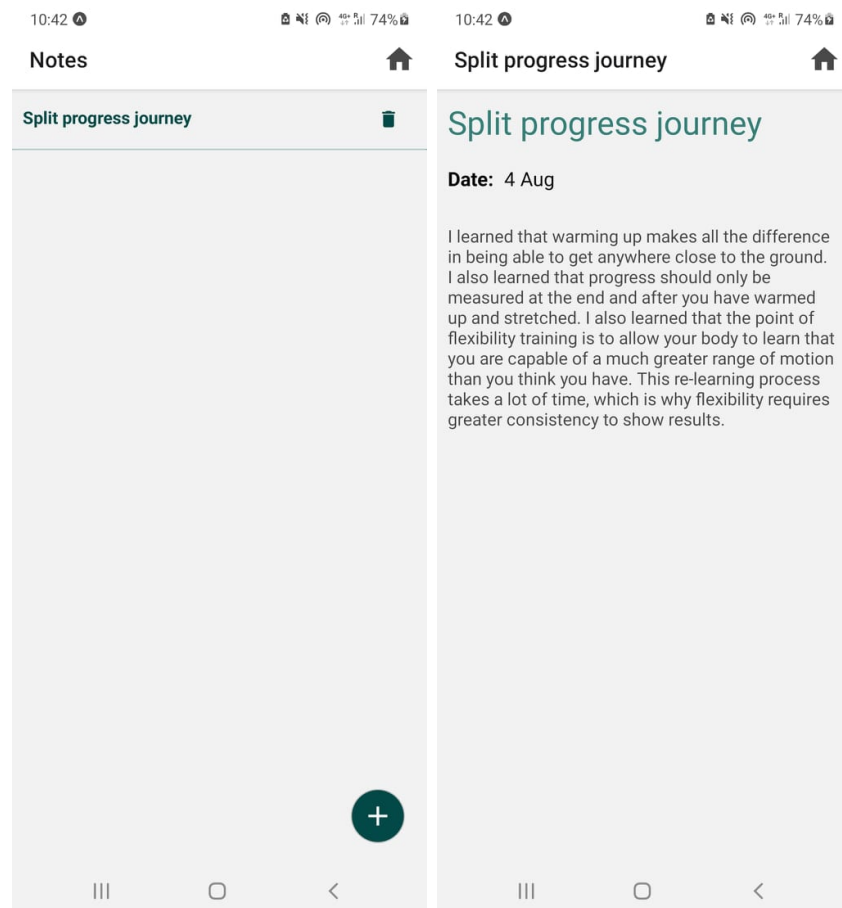


Slika 5.6: Prazna forma za stvarjanje zapiska (levo) in forma z vneseno vsebino zapiska (desno)

Po shranitvi zapiska je uporabnik samodejno preusmerjen nazaj na seznam zapiskov. Vsak zapisek v seznamu je 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 zapiske. S klikom na izbrani zapisek se odpre stran z naslovom zapiska, datumom, kdaj je bil zapisek ustvarjen in

zapisano besedilo uporabnika, (Slika 5.7).

V glavnem meniju se desno v glavi nahaja gumb za odjavo, v vsakem podmeniju pa se desno v glavi nahaja gumb Home, ki omogoča hitrejšo vrnitev v glavni meni.



Slika 5.7: Prikaz strani s seznamom zapiskov (levo) in prikaz strani z ogledom podrobnosti zapiska (desno)



## Poglavje 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 nadaljni 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 številen 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 avtentikacija uporabnika.

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

## 6.1 Ideje za nadaljni 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 spreminjal, 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] Asane. Dosegljivo: <https://www.vodenevadbe.si/post/kaj-pomeni-beseda-asana>. [Dostopano: 26. 5. 2021].
- [2] Expo. Dosegljivo: <https://docs.expo.io/>. [Dostopano: 25. 6. 2021].
- [3] Express. Dosegljivo: <https://expressjs.com/>. [Dostopano: 15. 7. 2021].
- [4] Git. Dosegljivo: <https://git-scm.com/>. [Dostopano: 15. 7. 2021].
- [5] O jogi. Dosegljivo: <https://www.hathajoga.si/joga>. [Dostopano: 20. 5. 2021].
- [6] Zgodovina joge. Dosegljivo: <https://www.hathajoga.si/zgodovina-joge>. [Dostopano: 20. 5. 2021].
- [7] Susan Hollister. *The Top 100 Best Yoga Poses Relieve Stress, Increase Flexibility, and Gain Strength*. Addison-Wesley, 2017.
- [8] Js. Dosegljivo: <https://www.theserverside.com/definition/JavaScript>. [Dostopano: 15. 7. 2021].
- [9] Mysql. Dosegljivo: <https://www.digitalocean.com/community/tutorials/what-is-mysql>. [Dostopano: 15. 7. 2021].
- [10] npm. Dosegljivo: <https://docs.npmjs.com/cli/v6/commands/npm>. [Dostopano: 25. 6. 2021].

- [11] Rn. Dosegljivo: <https://www.netguru.com/glossary/react-native>. [Dostopano: 25. 6. 2021].
- [12] Tableplus. Dosegljivo: <https://mariadb.com/kb/en/tableplus/>. [Dostopano: 15. 7. 2021].
- [13] Ts. Dosegljivo: <https://www.typescriptlang.org/>. [Dostopano: 15. 7. 2021].
- [14] Vsc. Dosegljivo: <https://www.educative.io/edpresso/what-is-visual-studio-code>. [Dostopano: 15. 7. 2021].
- [15] verywellfit. Dosegljivo: <https://www.verywellfit.com>. [Dostopano: 5. 6. 2021].