Elektrotehnički fakultet Verzija 1.0

Programiranje mobilnih uređaja

**DIPLOMSKI RAD**

**“RunPal”**

**Aplikacija za android OS mobilni uređaj**

Aleksa Vučković

2020/0035

**Istorija izmena**

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 21.2.2023. | 1.0 | Inicijalna verzija | Aleksa Vučković |
| 10.3.2023. | 2.0 | Dopune i ispravke | Aleksa Vučković |

Sadržaj

[1. Uvod 4](#_Toc160962311)

[2. Arhitektura sistema 4](#_Toc160962312)

# Uvod

Aplikacija RunPal namenjena je trkačima i omogućava merenje različitih parametara fizičke aktivnosti, njihovo pamćenje i pregled, uvid u korisne statističke podatke, ali takođe i povezivanje sa drugim korisnicima, organizovanje zajedničkih aktivnosti u kojima može učestvovati neograničen broj korisnika.

Aplikacija je namenjena mobilnim uređajima sa android operativni sistemo verzije 8.0 ili iznad. Izrađena je u Android Studio okruženju, na jeziku Kotlin koji se izvršava na Java virtuelnoj mašini, uz korišćenje Google Pixel 7 emulatora za testiranje.

# Arhitektura sistema

Osnovna komponenta sistema je aplikacija za android OS, koja se može instalirati na bilo kom android telefonu sa verzijom 8.0 i iznad.

Pored korisničke aplikacija, podaci se trajno čuvaju u nerelacionoj bazi podataka “MongoDB”, pokrenute preko MongoDB Atlas servisa (besplatna verzija). Zahteve za pristup podacima, kreiranje naloga i login opslužuje server koji je napisan u Node.js okruženju, a koristeži express.js biblioteku. Server se hostuje na “Heroku” platformi.

Pored objedinjene baze podataka, svaki uređaj ima lokalnu bazu podataka u vidu SQLite relacione baze.

Aplikacija se oslanja na android OS API visokog nivoa apstrakcije, kao i na Jetpack Compose bilbioteku za rad sa korisničkim interfejsom. Za komunikaciju sa serverom se koristi “Retrofit” biblioteka, a za pristup lokalnoj SQLite bazi “Room” bilbioteka.

# 3. Nalozi i autentikacija

Za korišćenje aplikacije potrebno je kreiranje naloga. Na serveru se lozinke kriptuju korišćenjem “bcrypt” algoritma, odnosno odgovarajućeg paketa za Node.js.

Prilikom uspešnog logovanja, server klijentu šalje JWT token koji traje 72 sata. Korišćenjem JWT (Json Web Token) tokena se postiže veći broj povoljnosti:

1. Korisnik ne mora često da unosi kredencijale. Tek ako nakon tri dana ne otvori aplikaciju, moraće ponovo da se uloguje.
2. U JWT “payload” segmentu čuvaju se neki od podataka koje bi server inače mora da dohvata iz baze.

Komunikacija sa serverom uspostavlja se kroz https protokol.

# 4. Sinhronizacija podataka između servera i lokalne baze

Aplikacija je namenjena i “onlajn” I “oflajn” upotrebi. Takođe, aplikacija nastoji da korisniku omogući jednak pristup svojim podacima sa više različitih uređaja. U skladu sa tim, podaci se sinhronizuju u oba smera. Podaci o aktivnosti se šalju na server u realnom vremenu ako je to moguće, ili naknadno. O nesinhronizovanim podacima se vodi evidencija u lokalnoj bazi podataka.

Slično, kada korisnik želi da sa novog uređaja pristupi svojim podacima, oni se preuzimaju sa servera i čuvaju lokalno, kako bi se nadalje omogućio “oflajn” režim rada.

Aplikacija je u toku aktivnosti otporna na gašenje procesa od strane operativnog sistema, pa i slučajnog unosa korisnika. Kada korisnik otvori aplikaciju, proverava se da li je neka aktvinosti prekinuta i to u lokalnom ali i nelokalnom repozitorijumu. Na osnovu toga se korisniku daje mogućnost da aktivnost nastavi.

# 5. Osnovne funkcionalnosti aplikacije

Aplikacija korisnicima nudi tri osnovne funkcionalnosti:

1. Aktivnost “solo” trčanja u kojoj se mere parametri poput razdaljine, brzine, potrošene energije, vremena, a dodatno se pamti i putnanja korisnika.
2. Aktivnost grupnog trčanja u kojoj može učestvovati do 5 korisnika. U zajedničim aktivnostima korisnici mogu da vide sve parametre aktivnosti drugih učesnika. Na primer, u toku aktivnost korisnik vidi putanju kretanja ostalih korisnika, njiihovu brzinu pređenu razdaljinu itd, a takođe može da vidi i neku vrstu rang liste.
3. Događaji su specijalna aktivnost koju kreiraju korisnici i zakazuju zao dređeno vreme, kao sa određenom dužinom. Događaji su javni u njima može da učestvuje neograničen broj korisnika. Na kraju događaja korisnici mogu da vide rang listu kao i svoj plasman na njoj. Korisnici mogu da pretražuju događaje, da ih prate, i dobijaju obaveštenja o njima.

Pored ovih funkcionalnosti, u aplikaciji korisnici mogu i:

1. Da vide detaljan pregled svake aktivnosti u vodu grafova i statističkih podataka, uključujući I celu istoriju prethodnih aktivnosti.
2. Da vide statističke podatke na nivou svih aktivnosti, u različitim vremenskim periodima.
3. Da menjaju podatke na svom nalogu, da odaberu željene merne jedinice, itd.

# 6. Detalji implementacije

Podaci o lokacije se dobijaju kroz FusedLocationProvider interfejs android razvojnog okruženja. FusedLocationProvider podatke o lokaciji računa na osnovu GPS podataka, Wi-Fi mreže i drugih izvora podataka. Iako bi korišćenje samo GPS signala dalo najpouzdanije rezulatate, FusedLocationProvider pruža zadovoljavujuću uslugu sa značajno manjom potrošnjom energije.

Dobijena lokacije se filtrira i prerađuje koristeći MovingAverageFilter klasu koja predstavlja jednostavnu implementaciju standardnog moving average filtera, i PostionFilter klase koja je dodatno prilagođena za podatke o lokaciji.

Na osnovu dobijenih lokacija, računa se najpre pređena razdaljina Haversine formulom, koja je aproksimativnog karaktera i podrazumeva Zemlju savršeno okruglog oblika, ali je za male razdaljine zadovoljavajuća:

Na osnovu vremene beleženja lokacija lako se računa brzina (koja se takođe dodatno filtrira).

Potrošnja energije u kalorijama po jedinici vremena računa se na osnovu potrošnje kiseonika (VO2), na osnovu formule iz sledećeg izvora <https://www.ideafit.com/wp-content/uploads/files/_archive/062005_calculatin.pdf>, odnosno na osnovu brzine, nagiba terena i težine subjekta. Potrošnja kiseonika u jedinicama l/s/kg (litar po sekundi po kilogramu) dobija se na osnovu:

gde je brzina izražena u m/s, a uspon kao odnos razlike u visini i pređene razdaljine. Kada se dobijena vrednost pomnoži sa težinom, dobijamo potrošnju kiseonika u litrima po sekundi, a na osnovu gore navedenog izvora, na svaki litar kiseonika potroši se aproksimativno 5 kilokalorija, pa se potrošnja energija u kilokalorijama po sekundi dobija na osnovu sledeće formule:

Primer: Korisnik ima 80kg, trči tempom 5min/km, odnosno 3.33 m/s, na usponu od 5% (na 10 m razdaljine visina se poveća za 0.5m). Tada je VO2 = 0.000817 l/s/kg, a PE = 0.327 kCal/s. Nakon 50 minuta, odnosno 10km, korisnik će potrošiti 980kCal.  
Drugi korisnik ima 60kg, trči tempom 6min/km, odnosno 2.78 m/s, na usponu od 0%. Tada je VO2 = 0.00056 l/s/kg, a PE = 0.167 kCal/s. Nakon 60 minuta, odnosno 10km, korisnik će potrošiti 600kCal.  
Ako isti korisnik trči na isponu od 10%, tada će za istu razdaljinu potrošiti 870kCal.

Zbog ovoga korisnik mora da unese i svoju težinu, a ovaj parametar može i da promeni na profilnoj stranici.

Za prikaz mape u aplikaciji korišćen je GoogleMaps API, koji pored samog prikaza omogućava i crtanje putanja, markera i upravljanje položajem kamere na mapi.

U toku grupnih aktivnosti, potrebno je sa servera dohvatati sveže podatke o ostalim korisnicima, a takođe u u toku događaja, kada se uživo prikazuju podaci o rangiranju. Razmatrana su dva pristupa:

1. WebSocket i njemu slični protokoli omogućavaju dvosmernu komunikaciju između servera i kljienta. Kljient ne mora da šalje zahtev serveru za ažuriranjem podataka, već čeka na nove podatke koje server sam šalje kljinetima koji te podatke prate. Ovaj pristup umanjuje količinu saobraćaja ali usložnjava implementaciju servera, kao i klijenta.
2. HTTP polling je stariji i jednostavniji pristup koji se oslanja na HTTP protokol. Klijent u redovnim intervalima šalje zahtev za novim podacima koristeći klasičan HTTP zahtev. Mana ovog pristupa je svakako veća količina saobraćaju i potencijalno manje ažurni podaci, ali on takođe ima i svoju prednost – kljient u potpunosti kontroliše tempo pristizanja novih informacija i može prozvoljno da odredi interval ili odloži dohvatanje nobih podataka.

U ovoj aplikaciji je primenjen princip HTTP polling.

Na kraju aktivnosti se podaci prikazuju numerički i u vidu grafika. Broj bilbioteka za crtanje grafova koje su prilagođene Jetpack Compose biblioteci je relativno mali, pa su za potrebe ovog projekta implementirane LineChart i ScatterChart metode za isrtavanje grafika.

Aplikacija koristi „string resource“ fajlove koji omogućavaju jednostavno prilagođavanje korisničkog interfejsa različitim govornim područjima. Podrazumevano se korisni engleski jezik, a dodatno postoji i resurs za srpski jezik.

# 7. Proširenja

Neka od mogućih proširnja aplikacije su:

1. Dodavanje podrške za povezivanje sa pametnim satovima, a pre svega sa onim koji koriste Google-ov WearOS.
2. Pored zajedničkih aktivnosti, treba omogućiti i interakciju između korisnika u vidu objava i poruka.
3. Dodatna podešavanja i opcije za konfiguraciju interfejsa.