

GoraNiNora: Kontekstno-odvisno obveščanje za varen obisk gora

Domen Vilar¹, Veljko Pejović^{1,2} and Bojan Blažica²

¹Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, Večna pot 113, 1000 Ljubljana, Slovenija

²Odsek za računalniške sisteme, Institut Jožef Stefan, Jamova 39, 1000 Ljubljana, Slovenija

Abstract. The growth of tourism also leads to the increase in mountaineering-based accidents. In this work we rely on mobile computing to improve the safety of mountain tourists. Through interviews and surveys we analyse the overall awareness and conclude that mountaineers often lack relevant practical knowledge. Thus, we develop a mobile application for context-dependent dissemination of educational content. The app takes into account the current location of a user as well as the detailed weather conditions when deciding which information to show to a user. The rules for which content to show to a user were devised in collaboration with mountaineering safety experts and the relevant literature. We implement the app in Android operating system, evaluate its usability and energy requirements, and demonstrate that our app represents an important tool for information dissemination when mountaineering.

Povzetek. Z razvojem turizma se vse bolj pogosto dogajajo nesreče, zlasti v gorah. V tem delu uporabljamo sodobne mobilne tehnologije, da bi prispevali k varnosti uporabnika v gorah. Preko intervjujev in anket smo izvedeli, kakšno je stanje in ugotovili, da je potreba po informiranju udeležencev v hribih, saj ti nimajo dovolj znanja. Zato smo se odločili razviti mobilno aplikacijo za kontekstno odvisno sporočanje, ki uporabniku podaja informativno-izobraževalne vsebine glede na trenutne razmere v gorah in njegovo trenutno lokacijo. Bolj ko je uporabnik izobražen o hoji v gore ter razmerah, manjša je verjetnost za nastanek nesreče. Iz intervjujev in pogovorov s strokovnjaki s področja varnosti v gorah ter strokovne literature smo določili pravila za prikaz opozoril. Aplikacijo smo razvili za mobilne naprave Android. Na koncu smo opravili še test uporabniške prijaznosti sistema, ki je pokazal, da naša aplikacija lahko predstavlja pomembno orodje za informiranje obiskovalcev gora ter ocenili energetske učinkovitost naše aplikacije.

Keywords

mountain safety, context-aware notification system, mobile application, Android

1. Uvod

Vedno več ljudi zadnje časa zahaja v hribe, saj išče oddih od vsakodnevnega hitrega življenja. Veliko ljudi zahaja v hribe tudi pozimi, ki prej nikoli niso, in zato nimajo ustreznih izkušenj. Ljudje, ki niso podučeni o hoji v hribe in nevarnostih, ki jim pretijo, lahko ogrozijo svoje življenje. Statistike nesreč Gorske reševalne zveze Slovenije (GRZS) [13] kažejo, da se je v zadnjih desetih letih število intervencij v gorah podvojilo. Če je bilo leta 2010 samo okoli 300 intervencij GRZS, jih je bilo v letu 2020 že več kot 600.

Z vzponom informacijsko-komunikacijskih tehnologij je trg ponudil kar nekaj rešitev, ki so poskušale zmanjšati število nesreč v gorah. Leta 2012 so se pojavile tri aplikacije: iSis, Snøg

Human-Computer Interaction Slovenia 2022, November 29, 2022, Ljubljana, Slovenia

✉ dv6526@student.uni-lj.si (D. Vilar); veljko.pejovic@fri.uni-lj.si (V. Pejović); bojan.blazica@ijs.si (B. Blažica)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



CEUR Workshop Proceedings (CEUR-WS.org)

in SnoWhere [12], ki so jih razvijalci oglaševali kot aplikacije, ki rešujejo življenje. Aplikacija je zaznala, ali je uporabnik udeležen v plazju, ter obvestila kontakte in pristojne službe na tistem območju. Te aplikacije uporabnika niso opozarjale pred nevarnostmi v gorah in ga izobraževale, vendar so ukrepale šele takrat, ko je uporabnik že bil v nevarnosti. Tega se je dobro zavedal Kanadski plazovni center, ki je zelo hitro izdal opozorilo in odsvetoval uporabo tovrstnih aplikacij namesto plazovne žolne [14]. Naslednje leto 2013 so se aplikacije umaknile s trga zaradi spoznanja, da so od njihovega pravilnega delovanja odvisna življenja ljudi.

Zgoraj omenjene rešitve se osredotočajo na reševanje ali pa so vir zelo splošnih izobraževalnih vsebin, katerih uporaba je pozimi v gorah nepraktična. Naš cilj je razviti vsestransko praktično orodje, ki bo izboljšalo varnost udeležencev v gorah. Za razliko od prejšnjih rešitev, naš pristop razvoja aplikacije izhaja iz celovite analize zahtev, ki temelji na odgovorih 950 anketiranih turnih smučarjev in gornikov ter na intervjujih s strokovnjaki iz Agencije Republike Slovenije za okolje (ARSO), Planinske Zveze Slovenije (PZS), Uprave Republike Slovenije za zaščito in reševanje (URSZR), z gorskimi reševalci in s planinskimi vodniki.

Analiza zahtev je pokazala, da je pomanjkanje ustreznih informacij v ključnih situacijah s katerim se obiskovalec gor sooča, eden izmed glavnih razlogov za nastanek nevarnih situacijah. Zato smo se odločili razviti aplikacijo **GoraNiNora** za mobilne pametne naprave, ki s pomočjo vgrajenih senzorjev avtomatsko ugotavlja kontekst, v katerem se uporabnik nahaja, in po potrebi dostavlja informativno-izobraževalne vsebine, ki pripomorejo k varnejšemu gibanju v gorah. Zaznavanje konteksta je v aplikaciji implementirano z upoštevanjem dobrih sodobnih praks mobilnega programiranja in zaznavanja, zato ima aplikacija minimalen vpliv na porabo energije mobilne naprave. Pravila, ki sprožijo interakcijo z uporabnikom, in vsebina poslanih opozoril upoštevajo strokovno znanje s področja gorništv. Posamezne dele aplikacije smo testirali z vidika porabe energije in uporabniške prijaznosti z desetimi prostovoljci.

Rešitev, ki smo jo razvili, je prvi prototip. V prihodnosti si želimo, da bi aplikacijo uporabljale tudi pristojne službe za varstvo pred snežnimi plazovi ali druge institucije, ki se ukvarjajo s snežnimi plazovi in nevarnostmi v gorah, zato da bi v aplikacijo vnašale znanje in pravila za dostavljanje obvestil. Ker želimo vključiti širšo skupnost v uporabo aplikacije, smo celotno kodo aplikacije objavili kot odprtokodni projekt na GitHub [6].

2. Analiza potreb

Preden smo se lotili razvoja aplikacije, smo naredili analizo potreb z morebitnimi uporabniki aplikacije. V nadaljevanju bomo predstavili analizo ankete in intervjujev, ki nam je bila v pomoč pri oblikovanju funkcijskih zahtev aplikacije.

2.1. Anketa

Na uspešnost razvojne rešitve vpliva tudi interakcija med uporabniki in razvijalci rešitve [7]. Za še boljši pregled navad uporabnikov o uporabi lavinskih naprav, njihovem izobraževanju na področju varnosti v gorah in opreme, s katero se odpravijo v hribe, smo izvedli spletno anketo. Za anketo smo uporabili spletno orodje 1KA ter jo objavili na spletni strani PZS in v dveh skupinah socialnega omrežja Facebook (*Turno smučanje - Snežak* in *Ideje kam v hribe*). Zbrali

smo 950 ustreznih odgovorov, ki jih bomo ovrednotili. Med anketiranci je bilo 72% moških in 28% žensk. Povprečna starost anketiranca je 36,9 let

Preko anket smo izvedeli, da so udeleženci v gorah pozimi dobro opremljeni. Lavinski trojček, ki ga sestavljajo lavinska žolna, lopata in sonda, je najpogosteje uporabljen med turnimi smučarji in manj med gorniki. Verjetno gorniki mislijo, da plazovi za njih ne predstavljajo tolikšne nevarnosti, kar je daleč od resnice, saj plazovi pretijo vsem obiskovalcem gora pozimi. Samo 55% jih vsako leto ponovi znanje o plazovni varnosti. Tisti, ki ga ponovijo, se najpogosteje udeležijo predavanj o plazovni varnosti in celodnevni delavnici na terenu. Menimo, da anketirancem primanjkuje znanja o plazovni varnosti, kar potrjujejo njihovi odgovori, da ne poznajo delovanja lavinske žolne, ki je nujna za uspešno reševanje, da vsako leto ne ponovijo znanja o plazovni varnosti ter svoje izkušnje v gorah povprečno ocenijo s 3,1 od 5. Kar 88% anketiranih pred vsako turo spremlja poročilo o nevarnosti snežnih plazov, kar pomeni, da so jim podatki o trenutnem stanju pomembni. 93% anketiranih v hribe hodijo v dvoje ali skupini, zato bi aplikacija lahko upoštevala, da uporabnik v hribe ne hodi sam. 64% anketiranih za sledenje aktivnosti uporablja mobilni telefon. To pomeni, da aplikacija ne bi dodatno porabljala energije za pridobivanje lokacije, saj ne bi bila edina, ki bi zahtevala lokacijo od sistema Android preko *Location API*-ja (Application Programming Interface - vmesnik uporabniškega programa) [8].

2.2. Intervjuji

Intervjuje smo izvedli s strokovnjaki s področja varnosti v gorah. Poiskali smo ljudi, ki se poklicno ukvarjajo z delom povezanim z gorami ali pa imajo večletne izkušnje z gorništvom. Namen intervjujev je bil, da čim več izvemo na področju varnosti, tehnologijah in na splošno o uporabi mobilnih naprav v gorah. Intervjuji so bili dolgi do ene ure. Pogovarjali smo se s strokovnjaki iz Agencije Republike Slovenije za okolje (ARSO), Planinske zveze Slovenije (PZS), Uprave Republike Slovenija za zaščito in reševanje (URSZR), vodjo Službe za varstvo pred snežnimi plazovi Zelenica-Tržič, dvema prostovoljnima gorskima reševalcema, planinsko vodnico in vodjo projekta Crossrisk [11].

Pristop k zmanjševanju nesreč v gorah z aplikacijo, ki naj bi nadomeščala žolno in obvestila kontakte v primeru nesreče ni pravilen, kar smo izvedeli iz intervjujev. Strokovnjaki so mnenja, da taka aplikacija uporabniku daje vtis varnosti, saj je uporabnik prepričan, da bo aplikacija v primeru plazu obvestila reševalne službe. Menijo, da ima mobilna naprava možnosti za izboljšanje varnosti v gorah, vendar moramo paziti, da aplikacija uporabniku ne bo dajala lažnega vtisa varnosti.

Ker imamo namen razviti aplikacijo za uporabo v gorah, je zelo pomembno vedeti, kakšne so omejitve pri uporabi mobilnih naprav v gorah pozimi. O uporabi mobilnih naprav pozimi smo povprašali intervjuvance. Zaradi nizkih temperatur in slabega signala v gorah se baterija hitreje sprazni. Mobilno napravo imamo ponavadi čim bližje telesu zaradi toplote in je zato nedostopna. Ravno tako imamo pozimi na rokah rokavice, kar oteži upravljanje z mobilno napravo. Zato mora biti čas interakcije z napravo čim krajši.

Deležniki bi rajši imeli aplikacijo z izobraževalnimi vsebinami, ki podaja znanje uporabnikom. Omenili so, da se jim zdi najpomembnejše izobraževanje ljudi, saj menijo, da tako lahko zmanjšajo nesreče v gorah. Učnih vsebin o varnosti v gorah je veliko, a so te zelo razpršene in jih morajo ljudje sami poiskati. Ker večina ljudi nima dovolj izkušenj in znanja, težko izlušči,

katero znanje je pomembno. Aplikacija bi podajala učne vsebine in informacije, ki so relevantne za trenutni prostor in čas.

2.3. Oblikovanje funkcijskih zahtev

Na podlagi intervjujev in analize ankete smo oblikovali funkcijske zahteve. Z uporabniškega vidika smo se osredotočili na funkcionalnosti, ki naj jih aplikacija nudi. Funkcionalne zahteve smo določili s primeri uporabe.

2.3.1. Prikaz splošnih in trenutnih opozoril

Glavni namen aplikacije je uporabnikom dostavljati kontekstno odvisne informacije pri obisku gora, pa tudi informacije bolj splošnega pomena, na podlagi katerih se uporabnik lahko odloči za obisk gora. Zato so opozorila glavni koncept aplikacije, ki so na voljo uporabniku. Aplikacija torej uporabniku omogoča pregled trenutno zaznanih in splošnih opozoril. Glede na izbrano območje aplikacija prikaže splošna opozorila. V implementirani različici aplikacije so območja, ki jih uporabnik lahko izbere: južni in zahodni Julijci, osrednji Julijci in zahodne Karavanke ter Kamniško-Savinjske Alpe in vzhodne Karavanke. S splošnimi opozorili želimo uporabnika vnaprej opozoriti na različne situacije v gorah. Na primer v primeru dežja ali novo zapadlega snega želimo uporabnika prej obvestiti, da lahko prilagodi prihajajoče zimske ture.

Aplikacija bo uporabniku omogočala tudi pregled *trenutnih opozoril*. Trenutna opozorila so prikazana samo takrat, ko aplikacija zazna, da se uporabnik nahaja v hribih. Aplikacija s pomočjo GPS (Global Positioning System - sistem globalnega pozicioniranja) zazna, na katerem od treh gorskih območij se nahajamo, nadmorsko višino, naklon, usmerjenost površja ter trenutno vremensko napoved in z ujemanjem vnaprej definiranih pravil prikaže trenutno aktualna opozorila. Ko aplikacija zazna hojo v hribe, se v aplikaciji prikaže obvestilo.

2.3.2. Avtomatsko zaznavanje konteksta

Aplikacija s pomočjo senzorjev samodejno zazna hojo, in v primeru, da se uporabnik nahaja v hribih, vklopi pošiljanje trenutnih opozoril. Če zaradi nenatančnih podatkov GPS ali senzorskih podatkov, vgrajenih v mobilno napravo, ta ne zazna hoje v hribe, lahko uporabnik ročno vključi pošiljanje trenutnih opozoril. Hoja v hribe se zazna, ko uporabnik prehodi 50 višinskih metrov v zadnjih 30 min. Če želi uporabnik takoj prejemati trenutna opozorila, lahko prejemanje le teh ročno vklopi v aplikaciji. V primeru, da je aplikacija napačno zaznala hojo v hribe, lahko uporabnik izkopi pošiljanje trenutnih opozoril za eno uro.

Aplikacija bo glede na zaznan kontekst uporabniku preko obvestil v sistemu Android sporočila trenutno zaznana opozorila. Ob kliku na obvestilo se odpre zaslon aplikacije, kjer si lahko uporabnik s premikanjem levo in desno po zaslonu pogleda trenutno zaznana aktualna opozorila.

2.3.3. Pregled zgodovine opozoril

Med zgodovino izdanih opozoril si bo uporabnik lahko pogledal, katera trenutna in splošna opozorila je aplikacija predlagala. Z zgodovino opozoril uporabnik vodi evidenco predlaganih

opozoril in si jih lahko še enkrat bolj podrobno pogleda. Tu se prikažejo trenutna in splošna opozorila starejša od enega dne.

2.3.4. Uporabnost v zunanem okolju

Želeli smo, da je aplikacija čimbolj uporabna, predvsem med samo uporabo v hribih pozimi, kjer zaradi dnevne svetlobe in težje berljivosti potrebujemo večje črke. Zaradi večjih črk smo morali prilagoditi tudi uporabniški vmesnik, saj bi v primeru daljšega besedila v sami aplikaciji ta postal zelo nepregleden.

3. Mobilna aplikacija GoraNiNora

Na podlagi funkcijskih zahtev smo razvili aplikacijo GoraNiNora, katere implementacija je opisana v tem poglavju.

3.1. Zaznavanje relevantnega konteksta

Mobilne naprave imajo vgrajeno množico senzorjev, kot so senzor pospeška, žiroskop in GPS, ki v kombinaciji s strojnim učenjem omogočajo, da naprava natančno ugotovi kontekst, v katerem se uporabnik nahaja. V naši aplikaciji je relevanten kontekst določen z lokacijo, trenutnim vremenom in plazovnimi razmerami na območju, kjer se uporabnik nahaja. Tabela 1 bolj točno opiše podatke, ki jih aplikacija zajema za določitev konteksta.

Tabela 1: Podatki, ki so na voljo za zaznavanje konteksta.

Podatek	Opis	Vir	Pogostost vzorčenja	Razpoložljivost
Nadmorska višina	Višina v m nad morjem	GPS	60 s, ko je uporabnik v hribih.	Pokritost z GPS signalom
Lokacija	Območje, kjer se nahajamo.	GPS	60 s, ko je uporabnik v hribih.	Pokritost z GPS signalom
Telesna dejavnost uporabnika	Telesna dejavnost, ki jo uporabnik izvaja: hoja ali mirovanje.	Activity Recognition Transition API [1]	Za vzorčenje skrbi API, na katerega nimamo vpliva, kako pogosto se podatki vzorčijo.	Vedno
Naklon	Naklon pobočja v stopinjah	ArcGIS API [3]	60 s, ko je uporabnik v hribih.	Pokritost z GPS signalom in internetna povezava
Ekspozicija	Usmerjenost pobočja v stopinjah (0 do 360)	ArcGIS API [3]	60 s, ko je uporabnik v hribih.	Pokritost z GPS signalom in internetna povezava
Temperatura	Temperatura odvisna od nadmorske višine	ARSO vreme [16]	Majhna (najmanj enkrat na dan)	Internetna povezava
Veter	Hitrost vetra odvisna od nadmorske višine	ARSO vreme [16]	Majhna (najmanj enkrat na dan)	Internetna povezava
Vreme	Oblačnost, vremenski pojav in intenzivnost padavin	ARSO vreme [16]	Majhna (najmanj enkrat na dan)	Internetna povezava

Problem	Plazovni problem v snežni odeji	ARSO plazovi [4]	Majhna (enkrat na dan)	Internetna povezava
Vzorec	Značilni plazovni vzorci	ARSO plazovi [4]	Majhna (enkrat na dan)	Internetna povezava
Nevarnost	Stopnja plazovne nevarnosti od 1 do 5	ARSO plazovi [4]	Majhna (enkrat na dan)	Internetna povezava

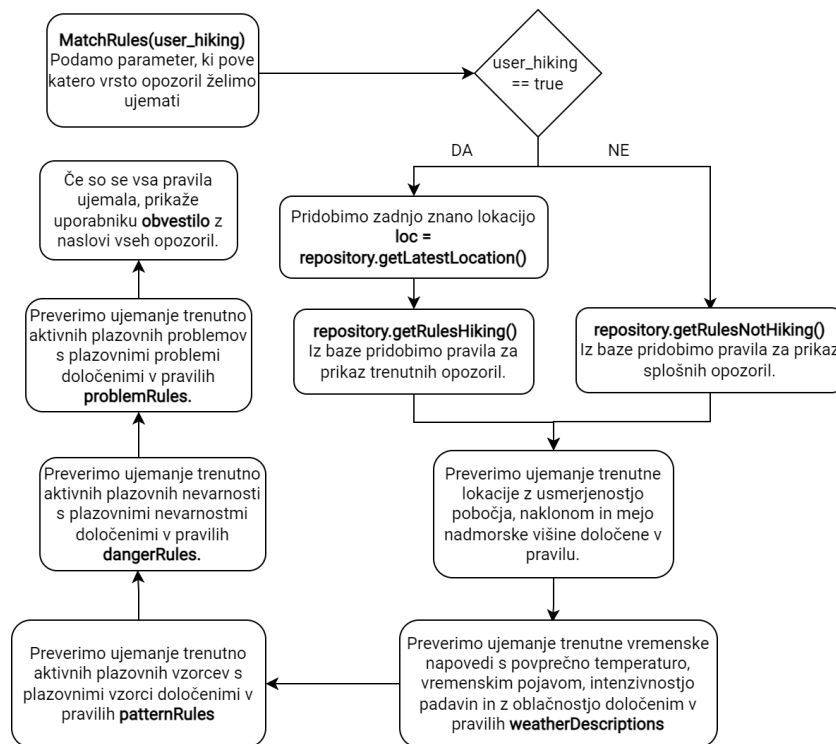
3.2. Določitev pravil za opozarjanje uporabnika

Pravila opisujejo, kateri pogoji morajo veljati, da uporabniku aplikacija prikaže določeno opozorilo. Opozorila ločimo na *trenutna* in *splošna* opozorila. Trenutna opozorila so kontekstno odvisna, kar pomeni, da pri prikazu trenutnega opozorila upoštevamo trenutno lokacijo, naklon, usmerjenost pobočja in nadmorsko višino. Kontekst lahko opišemo tudi z vremensko napovedjo, trenutno veljavnimi plazovnimi problemi, vzorci in nevarnostmi. Aplikacija s pomočjo trenutne lokacije uporabnika, trenutnega plazovnega biltena in vremenske napovedi zazna kontekst, v katerem se uporabnik nahaja in z ujemanjem vnaprej definiranih pravil za prikaz opozoril prikaže trenutno aktualna opozorila. Splošna opozorila so tista, ki jih uporabnik lahko pogleda vsakič, ko odpre aplikacijo, tudi če se ne nahaja v hribih. Glede na izbrano območje, ki ga uporabnik izbere s pomočjo spustnega menija, aplikacija prikaže splošna opozorila za izbrano območje. Območja, ki jih lahko uporabnik izbere so: južni in zahodni Julijci, osrednji Julijci in zahodne Karavanke ter Kamniško-Savinjske Alpe in vzhodne Karavanke. S splošnimi opozorili želimo uporabnika vnaprej opozoriti na različne situacije v gorah. Na primer v primeru dežja ali novo zapadlega snega želimo uporabnika prej obvestiti, da lahko prilagodi prihajajoče zimske ture.

Pravila za prikaz splošnih in trenutnih opozoril so shranjena v datoteki JSON (JavaScript Object Notation - objektna notacija za JavaScript) in so zapisana v podatkovno bazo ob prvem zagonu aplikacije. Ta pravila smo pridobili iz pogovorov s strokovnjaki in strokovne literature na področju varnosti v gorah. V trenutnem prototipu aplikacija ne bo pridobivala novih pravil iz zunanjega vira. Tako smo se odločili, ker gre za prototip aplikacije in je glavni namen preizkus podajanja kontekstno odvisnih informacij.

3.3. Ujemanje in prikaz pravil

Za prikaz določenega opozorila se morajo vsa določena polja v pravilu ujemanjati s trenutnimi podatki o lokaciji, vremenu in plazovnem biltenu. Za prikaz trenutnih opozoril se ujemanje pravil izvede vsakič, ko aplikacija zazna hojo v hribe in pridobi nov podatek o lokaciji. Za prikaz splošnih opozoril se ujemanje pravil izvede vsak dan, ko aplikacija pridobi sveže podatke o vremenu in plazovnem biltenu. Za ujemanje pravil skrbi razred `MatchRules`, ki mu podamo parameter `user_hiking`, ki pove, ali želimo izvesti ujemanje pravil za splošna ali trenutna opozorila. `MatchRules` shrani pravila, ki so se ujemala, v lokalno shrambo, zato da vodimo evidenco prikazanih opozoril in jih lahko prikažemo med zgodovino opozoril. Ko `MatchRules` pregleda vsa pravila, prikaže obvestilo za pravila, ki so se ujemala. Slika 1 prikazuje podrobnosti ujemanja pravil.



Slika 1: Podroben opis poteka ujemanja pravil

3.4. Uporabniški vmesnik

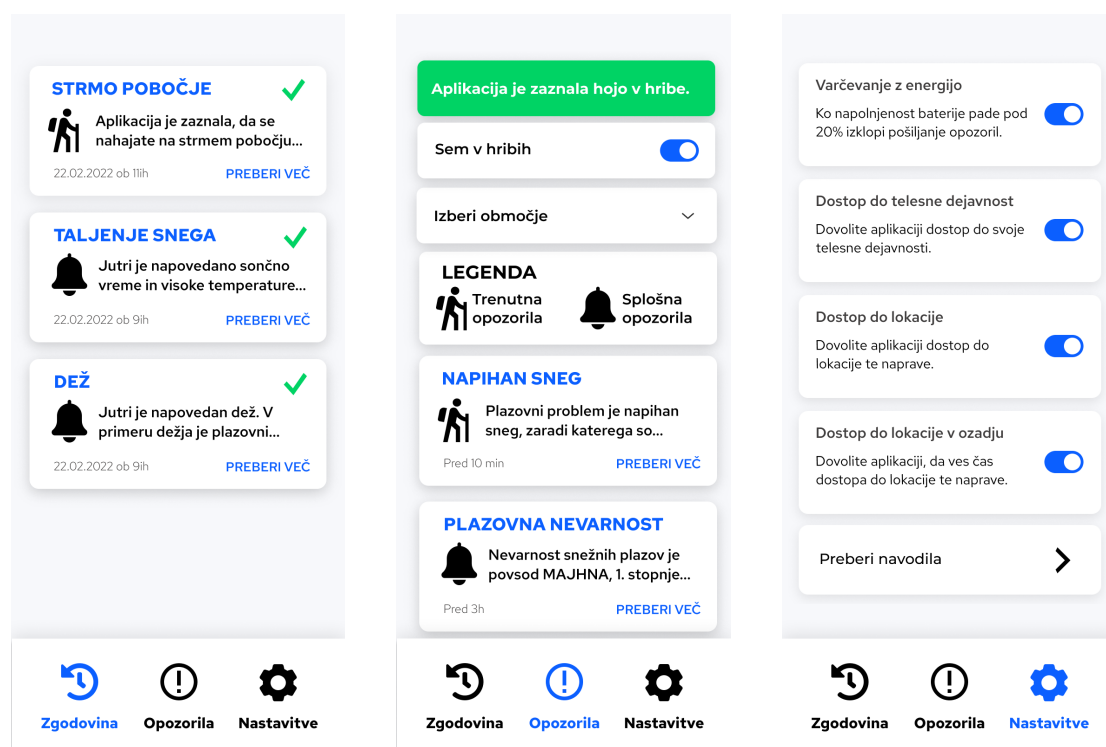
V skladu s najboljšimi praksami razvoja Android aplikacij smo uporabniški vmesnik zasnovali na fragmentih (angl. Fragment), ki jih ustvarjamo v okviru različnih aktivnosti (angl. Activity). Osnovni fragmenti aplikacije so prikazani na Sliki 2, podroben opis opozoril pa na Sliki 3.

Fragment za prikaz opozoril (Slika 2b) skrbi za prikaz trenutnih in splošnih opozoril. Tu prikažemo zaznana opozorila od trenutnega dne. Fragment prikazuje tudi informacijo o tem ali je hoja v hribe bila zaznana, ter omogoča ročno vklapljanje s pomočjo stikala. **Fragment za prikaz zgodovine** (Slika 2a) skrbi za prikaz trenutnih in splošnih opozoril, ki jih je aplikacija zaznala v preteklosti. Pretekla trenutna in splošna opozorila so tista, ki so starejša od enega dne. **Fragment za prikaz nastavitev** (Slika 2c) skrbi, da uporabnik lahko pogleda dovoljenja aplikacije, si ponovno prebere navodila in vključi varčevanje z energijo.

Aktivnost za prikaz trenutnih opozoril s klikom na obvestilo uporabniku sporoči trenutno zaznana opozorila preko obvestila v sistemu Android, na katerega uporabnik lahko klikne. Ob kliku na obvestilo se prikažejo trenutno zaznana opozorila (Slika 3). **Aktivnost za prikaz podrobnosti opozorila** skrbi za podroben opis opozorila, ko nanj kliknemo.

3.5. Uporaba zvoka in vibracije ob dostavi opozoril

Iz analize potreb smo izvedeli, da je lahko mobilna naprava med hojo pozimi nedostopna, saj želimo, da je čim bližje telesu na toplem. Tudi upravljanje z napravo je zelo oteženo zaradi uporabe rokavic. Zato bi želeli, da se ob zaznanih opozorilih sproži zvočni signal, ter vibracija, in tako aplikacija sporoči, da je zaznala potencialno nevarnost. Uporabnik aplikacije je obveščen o prispelem trenutnem opozorilu in si ga lahko prebere, ali, mu ga aplikacija sam prebere s pomočjo avtomatskega prevoda besedila v govor. V trenutni verziji aplikacije sta zvok in vibracija ob prihodu obvestila privzeti, kar pomeni, da uporabnik ne loči med prihodom obvestila od aplikacije GoraNiNora in drugih nameščenih aplikacij na sistemu. V prihodnosti nameravamo narediti analizo najbolj ustreznih načinov opozarjanja v kontekstu zimskega obiska gor.



(a) Zavihek zgodovina za prikaz zgodovine zaznanih trenutnih in splošnih opozoril.

(b) Zavihek opozorila za prikaz trenutnih in splošnih opozoril.

(c) Zavihek nastavitve za urejanje dovoljenj aplikacije, prikaz navodil in vklop varčevanja z energijo.

Slika 2: Uporabniški vmesnik aplikacije

4. Evalvacija

Testiranje je potekalo z desetimi prostovoljci (7 moških in 3 ženske), starimi od 15 do 65 let. Namen testiranja je bil ugotoviti, če je aplikacija dovolj enostavna in razumljiva za uporabo.



Slika 3: Prikaz trenutnih opozoril s klikom na obvestilo v sistemu Android

Testiranje uporabniške izkušnje in vmesnika je potekalo s pomočjo tokov v Figma [5], kjer elementom določimo interakcijo in prehode. Interakcije, ki smo jih uporabili, so klik na element in poteg levo ali desno po zaslonu. Prehod pomeni, da se ob interakciji premaknemo na drugo zaslonsko sliko. Pri testiranju smo vnaprej določili naslednje primere uporabe, ki jih morajo uporabniki uspešno opraviti:

1. *Navodila*: uporabnik naj prebere navodila aplikacije.
2. *Aplikacija v načinu, ko se nahajamo v hribih*: uporabniku povemo, da se trenutno nahaja v hribih in odpre aplikacijo. Preveri naj, če je aplikacija zaznala hojo v hribe. Prebere naj splošna opozorila za območje Kamniško-Savinjskih Alp.
3. *Aplikacija v načinu, ko se ne nahajamo v hribih*: uporabniku povemo, da se trenutno nahaja v hribih in odpre aplikacijo. Preveri naj, če je aplikacija zaznala hojo v hribe, in če ni, naj v aplikaciji vklopi, da se trenutno nahaja v hribih. Dovolj naj dostop do telesne dejavnosti in lokacije ter vklopi varčevanje z energijo.
4. *Klik na obvestilo*: klikne naj na obvestilo in prebere vsa opozorila.

Pravilno izvedene primere uporabe, odgovore na vprašanja ter dodatne komentarje smo si zabeležili. Na koncu smo jim zastavili še deset vprašanj iz vprašalnika splošne uporabniške prijaznosti sistema (SUS - System Usability Scale) [15].

Udeleženci testiranja so večinoma uspešno opravili zastavljene naloge. Samo enemu od

desetih testirancev ni uspelo ločiti med trenutnimi in splošnimi opozorili. Enemu pa ni uspelo preveriti, če se uporabnik nahaja v hribih, saj je spregledal modro obvestilo.

Na vprašanja o uporabnosti sistema so anketiranci odgovarjali s številko od 1 (sploh se ne strinjam) do 5 (se povsem strinjam). Po spodnji enačbi smo za vsakega testiranca izračunali oceno SUS. Z *vprasanje[i]* smo označili s katero številko (od 1 do 5) je testiranec ocenil *i*-to vprašanje.

$$Soda = 25 - \sum_{\substack{i=2 \\ \text{korak 2}}}^{10} vprasanje[i], \quad Liha = \left(\sum_{\substack{i=1 \\ \text{korak 2}}}^{10} vprasanje[i] \right) - 5$$
$$SUS \text{ ocena} = (Soda + Liha) * 2.5$$

Da smo dobili oceno SUS celotnega sistema, smo povprečili ocene SUS vseh testirancev. Ocena SUS je znašala 89.75 s standardnim odklonom 5.83. Glede na raziskave [9] ocena SUS nad 68 točk velja za nadpovprečno.

Rezultati evalvacije so pokazali možnosti za izboljšavo aplikacije. Legendo, ki prikazuje razliko med trenutnimi in splošnimi opozorili, so udeleženci raziskave poskusili klikniti, saj so mislili, da element omogoča filtriranje opozoril. Komentirali so, da bi si želeli funkcijo filtriranja trenutnih in splošnih opozoril. Drugi so komentirali, da bi rajši videli, da se med hojo v hribe prikažejo samo trenutna opozorila, ko ne hodi v hribe, pa samo splošna opozorila. Za implementacijo smo se odločili, da ne bomo imeli filtriranja opozoril, saj se najprej vedno prikažejo trenutna opozorila. Tudi zaradi lažjega rokovanja z rokavicami je bolje imeti čim manj elementov, na katere uporabnik klika.

Modro obvestilo, ki se prikaže, da je aplikacija zaznala hojo v hribe, ni bilo najbolj opazno, zato ga je eden udeleženec celo spregledal. Predlagal je drugo bolj opazno barvo. Predlog smo upoštevali in se odločili, da z zeleno barvo prikažemo obvestilo, da je aplikacija zaznala hojo v hribe.

Eden od testirancev je omenil, da bi zaupal opozorilom, če bi vedel, kdo je odgovoren za določitev in vsebino opozoril. Rešitev bi bila v navodilih ali nastavitvah navesti, kdo skrbi za določitev opozoril.

Na koncu smo ocenili še porabo energije. Orodje *Batterystats* [10], ki je že vključeno v ogrodje Android, zbira podatke o porabi energije na mobilni napravi. S pomočjo ukazne vrstice *adb* [2] smo prenesli poročilo o porabi energije *Batterystats* na računalnik. Orodje *Battery Historian* [10] pretvori poročilo v obliko HTML (Hypertext Markup Language - jezik za označevanje nadbisedila) za lažje analiziranje.

Predstavili bomo, koliko energije porabi aplikacija v približno eni uri delovanja. Aplikacijo smo namestili na mobilno napravo in vključili hojo v hribe. V tem načinu aplikacija pridobiva lokacijo GPS na 60 sekund. Za vsako novo pridobljeno lokacijo se kliče zunanji ArcGIS API [3] in `MatchRules`, ki preveri ujemanje s pravili za prikaz opozoril.

Aplikacija je bila aktivna 1 h 8 min 43 s. V tem času je 1 h 7 min 48 s uporabljala senzor GPS. V povprečju je prenesla 1.77 MB podatkov na uro. Povprečni čas uporabe procesorja na uro je bil 55 s uporabniškega časa in 19 s systemskega časa. `LocationUpdatesService` je bila aktivna 1 h 8 min. V tem času je samo naša aplikacija porabila 5.82 % od vse energije, ki je na voljo mobilni napravi. Izmerjeno predstavlja zgornjo mejo porabe energije, če bi se uporabnik nahajal v hribih.

5. Diskusija in zaključki

V tem delu smo naredili obsežno analizo možnosti za uporabo mobilnih tehnologij za izboljšanje varnosti v gorah. Ugotovili smo, da mobilne naprave mogoče niso najboljše orodje za zaznavanje nesreč v okolju, kot so hribi, vendar so zelo priročne za izobraževanje. Lahko v prihodnosti pričakujemo, da se to spremeni in bodo aplikacije zmožne dovolj natančno zaznati dejansko nevarnost? Z napredkom senzorjev v pametnih urah, ki so vse bolj pogoste, bomo lahko dobili še več informacij o trenutni situaciji. Mogoče v prihodnosti upravičeno pričakujemo, da bomo lahko sami s pomočjo obogatene resničnosti in strojnega učenja na napravi v realnem času izvedeli, katera pobočja so nevarna, da se jim bomo izognili.

Na podlagi analize potreb smo definirali funkcionalne zahteve in implementirali mobilno aplikacijo GoraNiNora za kontekstno-odvisno obveščanje uporabnikov. Naša aplikacija se zanaša na podatke iz senzorskih naprav in se s pomočjo vnaprej definiranih pravil odloča katero informacijo posredovati uporabniku. Podatke, ki jih naprava zbira na terenu, bi lahko uporabili tudi za učenje modela, ki bi lahko zaznaval tudi nove nevarne situacije. S pojavom zveznega učenja, ki omogoča manjšo porabo energije, napoved brez povezave z zunanjim strežnikom in krajši čas zakasnitve napovedi, bi zaznavanje nevarnih situacij lahko postalo možno.

Omenili bi, da seznam pravil, ki so trenutno določena, še ni kompleten. Tu pridejo na vrsto strokovnjaki, ki imajo ustrezno znanje za vnašanje pravil. Zaznavanje konteksta in dostava kontekstno odvisnih informacij se lahko uporabi na različnih področjih. Poleg gornišva bi aplikacija lahko prav prišla tudi v pomorstvu, jadrlnem padalstvu ali turizmu, seveda ob ustrezni prilagoditvi podatkov za zaznavanje konteksta in pravil za prikaz informacij.

V prihodnosti menimo, da bi za vnašanje opozoril v aplikacijo lahko skrbele institucije, ki se ukvarjajo z informiranjem o razmerah v gorah, kot sta PZS ali plazovne službe. Te imajo znanje, da poleg trenutno določenih pravil določijo še kopico ostalih.

Zahvala

Projekt "Kontekstno-odvisno približno računanje na mobilnih napravah" J2-3047 je sofinancirala Javna agencija za raziskovalno dejavnost Republike Slovenije iz državnega proračuna.

Literatura

- [1] *Activity Recognition API*. URL: <https://developers.google.com/location-context/activity-recognition>.
- [2] *Android Debug Bridge (adb)*. URL: <https://developer.android.com/studio/command-line/adb>.
- [3] *ArcGIS Summarize Elevation*. URL: <https://developers.arcgis.com/rest/elevation/api-reference/summarize-elevation.htm>.
- [4] *ARSO plazovi*. URL: <https://vreme.arso.gov.si/plazovi>.
- [5] *Figma*. URL: <https://figma.com>.
- [6] *GitHub projekt*. 2022. URL: <https://github.com/dv6526/GoraNiNora>.

- [7] Venkata N Inukollu in sod. "Factors influencing quality of mobile apps: Role of mobile app development life cycle". V: *International Journal of Software Engineering Applications (IJSEA)*, Vol.5, No.5, September 2014 (2014).
- [8] *Location Services*. URL: <https://developers.google.com/android/reference/com/google/android/gms/location/LocationServices>.
- [9] *Measuring Usability with the System Usability Scale (SUS)*. URL: <https://measuringu.com/sus/>.
- [10] *Profile battery usage with Batterystats and Battery Historian*. URL: <https://developer.android.com/topic/performance/power/setup-battery-historian>.
- [11] *Projekt Crossrisk*. URL: <https://crossrisk.zrc-sazu.si/>.
- [12] *Snowsafes Blog*. URL: <https://www.snowsafes.co.uk/avalanche-transceiver-app/>.
- [13] *Statistika nesreč*. 2022. URL: <https://www.grzs.si/resevanje/statistika-nesrec/>.
- [14] *There's not an app for that: CAC issues warning about avalanche apps*. 2013. URL: <https://newatlas.com/cac-warning-avalanche-search-apps/29575/>.
- [15] *Vprašalnik splošne uporabniške prijaznosti sistema*. URL: <https://hci.si/2015/02/12/article-sus-si/>.
- [16] *Vremenska napoved ARSO*. URL: <https://meteo.arso.gov.si/met/sl/service/>.