



УНИВЕРЗИТЕТ У НОВОМ САДУ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



УНИВЕРЗИТЕТ У НОВОМ САДУ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА

НОВИ САД

Департман за рачунарство и аутоматику

Одсек за рачунарску технику и рачунарске комуникације

ЗАВРШНИ (BACHELOR) РАД

Кандидат: Карталија Саша

Број индекса: Е12690

Тема рада: Праћење стања у саобраћају (бежично умрежавање возила)

Ментор рада: Самарџија др Драган

Нови Сад, Јун, 2012



КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:			
Идентификациони број, ИБР:			
Тип документације, ТД:	Монографска документација		
Тип записа, ТЗ:	Текстуални штампани материјал		
Врста рада, ВР:	Завршни (Bachelor) рад		
Аутор, АУ:	Саша Карталија		
Ментор, МН:	Самарџија др Драган		
Наслов рада, НР:	Праћење стања у саобраћају (бежично умрежавање возила)		
Језик публикације, ЈП:	Српски / латиница		
Језик извода, ЈИ:	Српски		
Земља публиковања, ЗП:	Република Србија		
Уже географско подручје, УГП:	Војводина		
Година, ГО:	2012		
Издавач, ИЗ:	Ауторски репринт		
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6		
Физички опис рада, ФО: (поглавља/страница/цитата/табела/слика/графика/прилога)			
Научна област, НО:	Електротехника и рачунарство		
Научна дисциплина, НД:	Рачунарска техника		
Предметна одредница/Кључне речи, ПО:	Андроид апликација, Стапаје у саобраћају, Надзор		
УДК			
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад		
Важна напомена, ВН:			
Извод, ИЗ:	У раду је описана реализација андроид апликације за бежично праћење стања у саобраћају, дојаве серверској страни о карактеристичним ситуацијама у саобраћајној мрежи, као и примање информација битних по учеснике у саобраћају, од стране сервера.		
Датум прихватања теме, ДП:			
Датум одбране, ДО:			
Чланови комисије, КО:	Председник:		Потпис ментора
	Члан:		
	Члан, ментор:		



KEY WORDS DOCUMENTATION

Accession number, ANO:		
Identification number, INO:		
Document type, DT:	Monographic publication	
Type of record, TR:	Textual printed material	
Contents code, CC:	Bachelor Thesis	
Author, AU:	Saša Kartalija	
Mentor, MN:	Dragan Samardžija, PhD	
Title, TI:	Condition monitoring of traffic (vehicle wireless networking)	
Language of text, LT:	Serbian	
Language of abstract, LA:	Serbian	
Country of publication, CP:	Republic of Serbia	
Locality of publication, LP:	Vojvodina	
Publication year, PY:	2012	
Publisher, PB:	Author's reprint	
Publication place, PP:	Novi Sad, Dositeja Obradovica sq. 6	
Physical description, PD: (chapters/pages/ref./tables/pictures/graphs/appendices)		
Scientific field, SF:	Electrical Engineering	
Scientific discipline, SD:	Computer Engineering, Engineering of Computer Based Systems	
Subject/Key words, S/KW:	Android application, Traffic condition, Monitoring	
UC		
Holding data, HD:	The Library of Faculty of Technical Sciences, Novi Sad, Serbia	
Note, N:		
Abstract, AB:	This paper describes the implementation of android applications for wireless monitoring of traffic, alert the server side of the characteristic situations in the traffic, and receive important information on road users, by the server.	
Accepted by the Scientific Board on, ASB:		
Defended on, DE:		
Defended Board, DB:	President:	
	Member:	
	Member, Mentor:	Menthor's sign

Zahvalnost

Zahvaljujem se svom mentoru Prof. Dr Draganu Samardžiji na stručnoj pomoći, strpljenju tokom izrade završnog (*bachelor*) rada i pomoći prilikom testiranja.

Srdačno se zahvaljujem kolegi Slavku Radonjiću koji je učestvovao u izradi projektnog zadatka.

Na kraju se zahvaljujem svima onima koji su na bilo koji način doprineli izradi ovog završnog rada.

SADRŽAJ

1.	Uvod	7
2.	Teorijske osnove	8
2.1	Android™ operativni sistem	9
2.2	TCP/IP model	11
2.2.1	TCP protokol.....	11
2.2.2	TCP portovi.....	12
2.2.3	IP protokol	13
2.3	Globalni pozicioni sistem	13
2.3.1	Osnove rada	13
2.3.2	Proračun pozicije.....	14
2.3.3	Preciznost i faktor greške.....	15
3.	Koncept rešenja	15
3.1	Automatska prijava lokacije	17
3.2	Ručna prijava događaja	17
3.3	Automatska detekcija neravnina	18
3.4	Pozadinski alarm.....	19
3.5	TCP komunikacija	19
4.	Programsko rešenje	21
4.1	RoadConditionActivity klasa	22
4.2	ReportActivity klasa	24
4.3	MenuActivity klasa.....	25
4.4	AccidentActivity klasa	26
4.5	CameraActivity klasa.....	28
4.6	HazardActivity klasa.....	31
4.7	JamActivity klasa.....	34
4.8	PoliceActivity klasa	35
4.9	IpMessage klasa.....	37
5.	Rezultati	38
6.	Zaključak	39
7.	Literatura.....	40

SPISAK SLIKA

Slika 1. – Izgled sistema	8
Slika 2. – TCP/IP model	11
Slika 3. – TCP portovi.....	12
Slika 4. – Koncept rešenja	16
Slika 5. – Scenario automatske prijave lokacije	20
Slika 6. – Scenario manuelne prijave događaja	18
Slika 7. – Alarm na udaljenosti.....	19
Slika 8. – Scenario uspešnog prijema alarma	21
Slika 9. – Prikaz klase RoadConditionActivity	23
Slika 10. – Početni ekran aplikacije	24
Slika 11. – Prikaz klase ReportActivity.....	25
Slika 12. – Izgled REPORT aktivnosti.....	26
Slika 13. – Prikaz klase MenuActivity	26
Slika 14. – Izgled MENU aktivnosti	27
Slika 15. – Prikaz klase AccidentActivity	27
Slika 16. – Izgled Accident aktivnosti.....	28
Slika 17. – Prikaz klase CameraActivity	29
Slika 18. – Izgled Camera aktivnosti.....	30
Slika 19. – Prikaz klase HazardActivity	31
Slika 20. – Prikaz izgleda aktivnosti Hazard	33
Slika 21. – Opcije u meniju dugmeta Weather, On Road, Shoulder	33
Slika 22. – Prikaz klase JamActivity	34
Slika 23. – Izgled Traffic jam aktivnosti	35
Slika 24. – Prikaz klase PoliceActivity	36
Slika 25. – Izgled Police aktivnosti	37
Slika 26. – Prikaz klase IpMessage	37

SPISAK TABELA

Tabela 1. – Format poruka.....	20
Tabela 2. – Prikaz klase projektnog zadatka.....	23

SKRAĆENICE

GPS – *Global Positioning System*, Globalni pozicioni sistem

OS – *Operating System*, Operativni sistem

TCP – *Transmission Control Protocol*, Protokol kontrole toka

IP – *Internet Protocol*, Internet protokol

FTP – *File transfer protokol*, Protokol slanja

HTTP – *HyperText Transfer Protocol*, Mrežni protokol

1. Uvod

U okviru rada potrebno je realizovati aplikaciju za android pametne uređaje. U ovu grupu uređaja spadaju svi oni koji kao operativni sistem koriste android, a pored toga imaju pojedine senzore koji su neophodni za potpuno iskorišćenje mogućnosti aplikacije (senzor pokreta (eng. ACCELEROMETER), GPS primopredajnik itd. Uređaju koji koristi ovu aplikaciju potrebna je stalna konekcija na internet da bi bio u mogućnosti da komunicira sa serverom.

Aplikacija je prvenstveno namenjena učesnicima u saobraćaju, praćenju stanja u saobraćaju i putnoj infrastrukturi. Samim tim je akcenat na sigurnosti učesnika u saobraćaju, kvalitetnijem životu naših vozača, bezbrižnjem i modernijem dostizanju cilja.

Elementi koji su realizovani u radu su sledeći:

- ✓ Praćenje policijskih patrola,
- ✓ Dojava o saobraćajnoj nesreći na putu,
- ✓ Dojava o postavljenoj kamери koja prati prolazak kroz crveno svetlo na semaforu,
- ✓ Dojava o uređaju koji prati brzinu kretanja vozila,
- ✓ Praćenje zagušenja na pojedinim saobraćajnicama,
- ✓ Opasnost na putu,
- ✓ Opasnost pored puta,
- ✓ Dojava o lošim vremenskim uslovima na putu.

Potpuna realizacija sistema prikazana je na slici 1.



Slika 1. – Izgled sistema

Sa jedne strane je uređaj koji koristi android aplikaciju, a sa druge strane je server, koji opslužen informacijama od strane korisnika aplikacije, daljom interakcijom komunicira sa svojim klijentima. Sa jedne strane imamo klijente koji su zapravo korisnici aplikacije, a sa druge strane imamo klijente koji putem internet pretraživača mogu dobiti željene informacije o putnoj infrastrukturi, kao i o dešavanjima na istoj.

2. Teorijske osnove

2.1 Android™ operativni sistem

Android™ operativni sistem je trenutno naj rasprostranjeniji operativni sistem za mobilne telefone, koji je zasnovan na *Linux* kernel i *GNU* softveru. U početku, ovaj OS je razvila firma Android Inc. (koju je posle kupio Google) i kasnije proširen na *Open Handset Alliance*.

Prilagođen je tako da se može koristiti na većini mobilnih uređaja, uključujući pored mobilnih telefona i tablet računare, laptop računare, netbook računare, smartbook računare, čitače elektronskih knjiga, pa čak od skoro i ručne satove.

Iako je android, *Linux* distribucija, on po nekim stvarima odstupa od većine standardnih *Linux* distribucija. Android nema standardni *X Window System* ili standardni skup *GNU* biblioteka pa tako nije u mogućnosti da pokreće aplikacije razvijene za druge standardne *Linux* sisteme.

Pokretanje samih aplikacija se, kod android-a, ne vrši direktno, već se aplikacije pokreću u okruženju odvojenom od ostatka sistema gde dobijaju samo određeni deo sistemskih resursa, pa tako nemaju pristup delovima sistema koji su im nepotrebni, što donekle poboljšava sigurnost i stabilnost sistema. Takođe pri instalaciji aplikacija korisnik dobija listu svih dozvola koje jedna aplikacija zahteva da bi se instalirala, što korisniku daje mogućnost da uoči potencijalno štetne aplikacije i obustavi njihovu instalaciju pre nego što dođe do oštećenja.

Sa tehničke strane, android predstavlja *Linux* operativni sistem razvijen za ARM i x86 arhitekturu i sastoji se od modifikovanog monolitnog *Linux* kernela zaduženog za podršku hardvera i funkcija niskog nivoa, skupa biblioteka zaduženih za dodatne podrške kao što su iscrtavanje grafike, podrška za dekodovanje video snimaka, podrška za SSL enkripciju itd...

U sklopu biblioteka se nalazi i odvojeni *Android Runtime* koji sadrži osnovne, bazne, biblioteke i *Dalvik* virtualna mašina zadužena za pokretanje aplikacija višeg nivoa napisanih u java programskom jeziku. Na višem nivou od biblioteka su sistemske aplikacije neophodne za upotrebu sistema od strane korisnika i tu se nalaze, *window manager*, menadžer resursa, menadžer instalacionih paketa, kao i aplikacije zadužene za obavljanje osnovnih funkcija vezanih za mobilne telefone ili uređaj na kom je instaliran android operativni sistem. Na najvišem nivou se nalaze krajnje korisničke aplikacije, odnosno aplikacije koje direktno koristi korisnik.

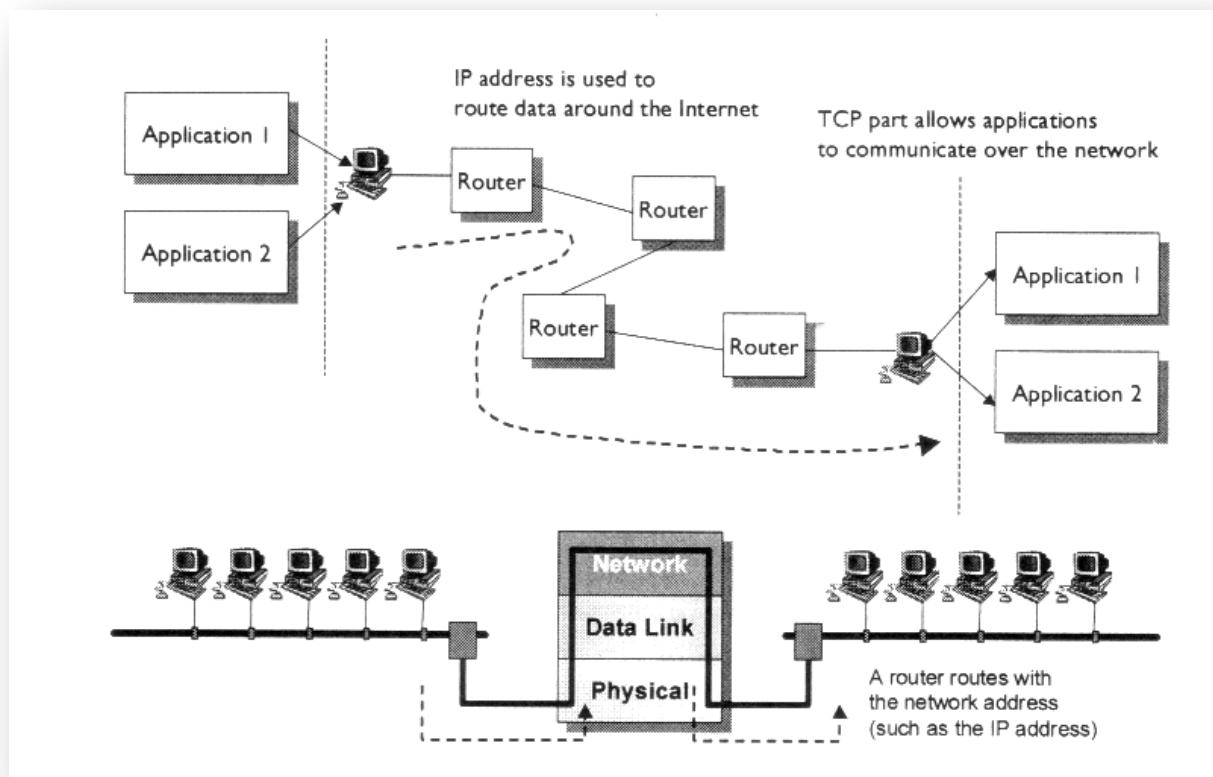
Ovakva arhitektura sistema nije iznenadjujuća jer predstavlja standardnu arhitekturu *Linux* sistema gde su segmenti sistema razdvojeni po nivoima na kojim rade. Za crtanje 3D grafike android koristi biblioteku zasnovanu na *OpenGL ES 2.0* specifikaciji, što ovom sistemu daje mnoge napredne grafičke sposobnosti. Android poseduje i ugrađenu podršku za *multitasking*.

Kroz svoju istoriju Android je imao nekoliko verzija od kojih je svaka donosila neku novinu i poboljšanje, tako je npr. verzija 1.0 bila prva zvanično dostupna verzija android operativnog sistema, V1.5 *Cupcake* je bila njena nadogradnja zasnovana na *Linux* kernelu 2.6.27, V1.6 *Donut* je koristio 2.6.29 *Linux* kernel i imao još više dodatnih mogućnosti u odnosu na prvu 1.0 verziju. Sa pojmom verzija 2.0 i 2.1 pod nazivom Eclair ispravljene su mnoge postojeće greške u samom sistemu i dodate dodatne podrške za rad sa kamerom, kao i poboljšana virtualna tastatura. Verzija 2.2 *Froyo* je prešla na novi kernel 2.6.32, ubrzala je rad sa memorijom i poboljšala performanse samog sistema, V2.3 *Gingerbread* takođe prelazi na novi kernel 2.6.35 i dodatno poboljšava korisnički interfejs, takođe donosi sa sobom i podršku za veće displeje kao i za neke dodatne senzore. Verzija 3.0 poznata i kao *Honeycomb* bila je zasnovana na kernelu 2.6.36 i bila je prilagođena tablet računarima, dodati su joj interfejs elementi kao što su *system bar* i *action bar* koji su prilagodeni za tablet računare, takođe pojednostavljena je i upotreba *miltitasking*, redizajnirana je i virtualna tastatura tako da omogući lakše i brže kucanje a uklonjeni su i neki sigurnosni propusti.

Verzija 4.0 koja nosi naziv *Ice Cream Sandwich* je trenutno najnovija verzija android operativnog sistema.

Ova verzija donosi redizajnirani korisnički interfejs prilagođen za displeje visoke rezolucije i dizajniran je tako da omogući čist i jasan prikaz teksta i grafičkih elemenata na ovakvim displejima.

2.2 TCP/IP model



Slika 2. – TCP/IP model

2.2.1 TCP protokol

TCP/IP (eng. Transmission Control Protocol / Internet Protocol) je komunikacioni protokol za Internet. TCP/IP definiše pravila koje računar mora koristiti za komunikaciju sa drugima. TCP/IP definiše kako uređaji trebaju biti povezani na internet i kako predaja i prijem podataka treba između njih da se odvija.

Internet predstavlja mrežu više desetina miliona računara koji su međusobno povezani na različite načine. Svi ti računari, korišćenjem TCP/IP protokola međusobno komuniciraju.

TCP/IP skup protokola prihvaćen je kao standard zbog pogodnosti koje je jedini u datom trenutku nudio, a neke od njih su:

- ✓ Nezavisnost o tipu računarske opreme i operacijskih sistema , te nezavisnost o pojedinom proizvođaču,
- ✓ Nezavisnost o tipu mrežne opreme u fizičkom pogledu i prenosnog medijuma , što omogućava integraciju različitih tipova mrež (Ethernet, Token Ring, X.25...),

- ✓ Jedinstveni način adresiranja koji omogućava povezivanje i komunikaciju svih uređaja koji podržavaju TCP/IP,
- ✓ Standardizovani protokoli viših nivoa komunikacijskog modela, što omogućava široku primenu mrežnih usluga.

TCP garantuje pouzdanu isporuku podataka u kontrolnom redosledu od izvora prema odredištu. Osim toga, TCP pruža i mogućnost višestrukih istovremenih veza prema jednom serveru od strane više klijenata, gde su najčešći primeri za to globalni internet ili poslužitelji elektronske pošte.

TCP je odgovoran za podelu podatka na IP pakete pre slanja i ponovnog povezivanja IP paketa u celinu pri prijemu. Korekciju podatka pri prenosu na internet/intranet obavlja TCP protokol.

Ako jedna aplikacija želi da komunicira sa serverom putem TCP-a ona šalje komunikacioni zahtev. Ovaj zahtev mora biti poslan na tačno određenu adresu. Posle uspostave veze između aplikacije i servera, TCP će zauzeti komunikacionu liniju između ova dva uređaja sve dok je ne zatvori jedan od njih.

2.2.2 TCP Portovi

+	Bitovi 0 - 3	4 - 9	10 - 15	16 - 31
0	Izvořni port			Odredišni port
32	Broj sekvence			
64	Broj potvrde			
96	Podatkovni ofset	Rezervirano	Zastavice	Prozor
128	Checksum			Hitni pokazivač
160	Opcionalno			
192	Opcije (nastavak)			Padding (do 32)
224	Korisnički podaci			

Slika 3. – TCP portovi

TCP upotrebljava određen raspon portova kojima razdeljuje primenske programe na strani pošiljatelja i primaoca. Svaka strana TCP konekcije ima dodeljenu 16-bitnu oznaku za obe strane aplikacije (slanje, prijem). Portovi su u osnovi podeljeni u 3 kategorije: poznati portovi, registrovani portovi i dinamički/privatni portovi.

Opšte poznati portovi (eng. *well known ports*) dodeljeni su od strane *Internet Assigned Numbers Authority*, organizacije koja se brine za IP adresni prostor, vršne domene te druge detalje vezane uz IP protokol. Ovi portovi su najčešće korišteni od strane sistemskih procesa,

koje koriste poznate aplikacije kada primaju konekcije pasivno slušajući promet na tim portovima. Neki primeri opšte poznatih portova su: FTP (TCP port 21), Telnet (23), SMTP (25) i HTTP (80).

Registrirani portovi se koriste kod aplikacija krajnjih korisnika kao izvorišni portovi prilikom konekcije poslužitelja, kao i za identifikaciju servisa registrovanih od trećih strana.

Dinamički/privatni portovi se takođe koriste na strani aplikacija krajnjih korisnika, ali nešto ređe. Dinamički/privatni portovi imaju samo lokalno značenje za određenu TCP konekciju.

TCP protokol definiše raspon portova od 0 do 65535, tj. ukupno ima 65536 mogućih različitih portova.

2.2.3 IP protokol

IP je odgovoran za slanje i prijem paketa podataka na internetu.

IP ne zauzima komunikacionu liniju između dva računara. IP smanjuje potrebu za mrežnim linijama. Svaka linija može biti korištena između više različitih računara u isto vreme. Sa IP se izvodi deljenje poruka (ili drugih podataka) u male nezavisne pakete i slanje između računara putem interneta. IP je odgovoran za "RUTIRANJE" svakog pojedinačnog paketa na tačno odredište.

2.3 Globalni pozicioni sistem

Globalni pozicioni sistem (eng. *Global Positioning System - GPS*) je trenutno jedini potpuno funkcionalan globalni satelitski navigacioni sistem (eng. *Global Navigation Satellite System*). GPS se sastoji od 24 satelita raspoređenih u orbiti Zemlje, koji šalju radio signal na površinu Zemlje. GPS prijemnici na osnovu ovih radio signala mogu da odrede svoju tačnu poziciju - nadmorsku visinu, geografsku širinu i geografsku dužinu - na bilo kom mestu na planeti danju i noću, pri svim vremenskim uslovima.

GPS ima veliku primenu kao globalni servis u raznim oblastima, u komercijalne i naučne svrhe: navigacija na moru, zemlji i u vazduhu, mapiranju zemljišta, pravljenju karata, određivanju tačnog vremena, otkrivanju zemljotresa i slično.

GPS je razvijen od strane Ministarstva Odbrane Sjedinjenih Američkih Država. U početku je korišćen isključivo u vojne svrhe da bi kasnije bio besplatno stavljen na raspolaganje svima kao javno dobro. Godišnji troškovi održavanja sistema su oko 750 miliona američkih dolara.

2.3.1 Osnove rada

GPS prijemnik je uređaj koji proračunava svoju poziciju na osnovu merenja udaljenosti od tri ili više GPS satelita. Svaki satelit emituje mikrotalasnu sekvencu radio signala koja je poznata prijemniku. Dok prijemnik prima taj signal, u stanju je da odredi vreme koje protekne od emitovanja radio signala sa satelita do prijema na svojoj poziciji. Udaljenost prijemnika od satelita se proračunava na osnovu tog vremena, budući da radio signal putuje poznatom brzinom. Signal takođe nosi informaciju o trenutnom položaju satelita sa kog se emituje. Ako se zna udaljenost prijemnika od satelita i pozicija satelita, poznato je da se prijemnik nalazi negde na sferi određene dimenzije u čijem je centru satelit. Pošto su poznate pozicije tri satelita i udaljenost prijemnika od svakog od njih, postupkom trilateracije se može odrediti pozicija prijemnika. Trilateracija se bazira na činjenici da se tri sfere seku u najviše dve tačke (od kojih jedna obično nema smisla).

Ovaj princip rada podrazumeva da su časovnici na svim satelitima, kao i na prijemniku potpuno sinhronizovani, da bi se vremenski razmak između poznate sekvence signala sa satelita i na prijemniku tačno izmerio. Na satelitima se nalaze atomski časovnici, veoma precizni i skupi. Međutim, prijemnik ima daleko manje precizan časovnik, kristalni oscilator. Nedostatak preciznosti se rešava uvodenjem merenja udaljenosti od još jednog satelita. Sat na prijemniku uvodi istu vremensku i prostornu grešku kada proračunava udaljenost od sva četiri satelita. Može se izračunati za koliko treba korigovati sat da bi se četiri sfere sekle u jednoj istoj tački. Na taj način se sat na prijemniku neprekidno koriguje. Jedna od primena GPS-a je veoma precizno računanje vremena i sinhronizacija časovnika.

2.3.2 Proračun pozicije

Geografske koordinate prijemnika se proračunavaju bazirano na Svetskom geodetskom sistemu, WGS84. Za početak, GPS prijemnici sa satelita neprekidno primaju navigacionu poruku koja u sebi sadrži informaciju o njihovoj poziciji.

Prijemnik identificuje signal sa svakog pojedinog satelita prema njegovoj jedinstvenoj digitalnoj sekvenci, pa meri razmak između vremena kada je signal emitovan i vremena kada je signal primljen. To se radi tako što prijemnik interno generiše signal sa istom digitalnom sekvencom kao što je ima signal sa satelita. Zatim polako menja vremensku fazu tog signala sve dok se interni signal i signal sa satelita ne podudare. U trenutku podudaranja, pomerena vremenska faza internog signala je jednaka vremenu potrebnom da signal putuje od satelita do prijemnika, na osnovu čega se može izračunati udaljenost prijemnika od satelita, s obzirom na poznatu brzinu kojom radio signal putuje. Ova udaljenost se naziva pseudoudaljenost. Pseudo

je zbog toga što je u ovom računanju prepostavljen da je interni časovnik prijemnika tačan, ali on sadrži izvesnu nepreciznost.

GPS prijemnik u svakom trenutku može da izračuna pseudoudaljenost od četiri satelita. Možemo da zamislimo četiri sfere od kojih svaka ima centar u po jednom od tih satelita a poluprečnik joj je udaljenost od tog satelita do prijemnika. To su četiri sfere koje se sve sekut u jednoj tački. Pošto signali sa svakog satelita putuju istom brzinom, u svakoj od pseudoudaljenosti je uračunata ista apsolutna greška. Kada bi sfere za poluprečnike imale pseudoudaljenosti umesto stvarnih udaljenosti, one se ne bi sekle u istoj tački, već bi sve bilo malo pomereno. Malom korekcijom pseudoudaljenosti za istu vrednost možemo podešiti da se sfere sekut u istoj tački. Kada se izračuna kolika je apsolutna greška u izračunavanju pseudoudaljenosti, onda se zna i kolika je nepreciznost internog časovnika prijemnika i on se podešava da tačnije pokazuje vreme. Ovo podešavanje se stalno dešava u vremenu.

Proračun pozicije na osnovu P-koda je konceptualno sličan, pod pretpostvkom da se signal može dekodovati. Šifrovanje ovog signala je zaštitni mehanizam. Ako se signal može uspešno dešifrovati, onda se može prepostaviti da je zaista poslat sa GSP satelita. U poređenju sa P-kodom koji se koristi u vojne svrhe, C/A kod je veoma osetljiv na ometanja. Pošto su digitalne sekvence GPS signala javno poznate, moguće je namerno ih emitovati generatorima signala.

2.3.3 Preciznost i faktori greške

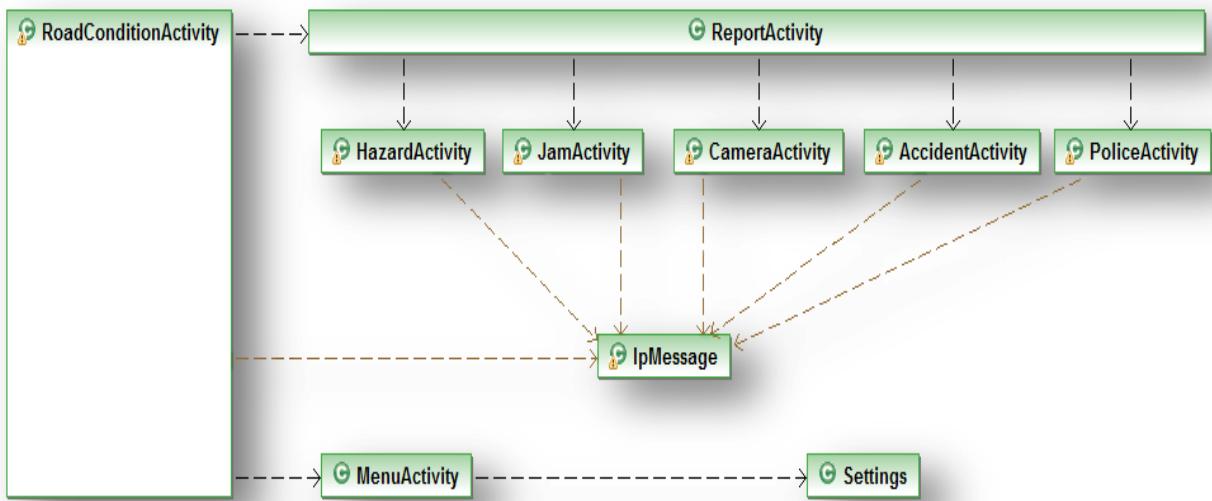
Za izračunavanje pozicije prijemnika koristi se tačna pozicija satelita i tačno vreme između emitovanja i prijema signala. Pošto se to vreme ustanavljava poređenjem signala sa satelita i internog signala, i u cilju poređenja se identificuju podižuće i spuštajuće ivice digitalnog signala, sadašnja elektronika u to unosi nepreciznost od oko 10 nanosekundi u C/A kod, što odgovara grešci od 3m u merenju udaljenosti. Kada bi pozicija satelita i vreme časovnika bilo apsolutno tačno, 3m bi još uvek bila najmanja greška na koju treba računati. Pošto je digitalni signal P-koda brži, greška koja se na ovaj način unosi je manja i iznosi samo 30 cm.

Ostali izvori nepreciznosti su atmosferski efekti koji utiču na brzinu prostiranja radio-signala, višestruke putanje signala, nepreciznost satelitskih časovnika, nepreciznost podataka o poziciji satelita, numeričke greške pri izračunavanju, brzina satelita i gravitacija Zemlje. Kada se svi ovi faktori zajedno uzmu u obzir, i pored metoda korekcije, ukupna greška određivanja pozicije je oko 15 m. Koristeći samo jedan prijemnik, bez obzira na njegov kvalitet i deklarisane tehničke karakteristike, nije moguće postići bolje položajnu tačnost od 50 metara u 66% slučajeva ili 100 metara u 99% slučajeva.

3. Koncept rešenja

Realizacija projektnog zadatka podeljena je u nekoliko zasebnih celina, koje se izvršavaju nezavisno jedna od druge. Veći deo funkcionalnosti sistema čine sledeće celine:

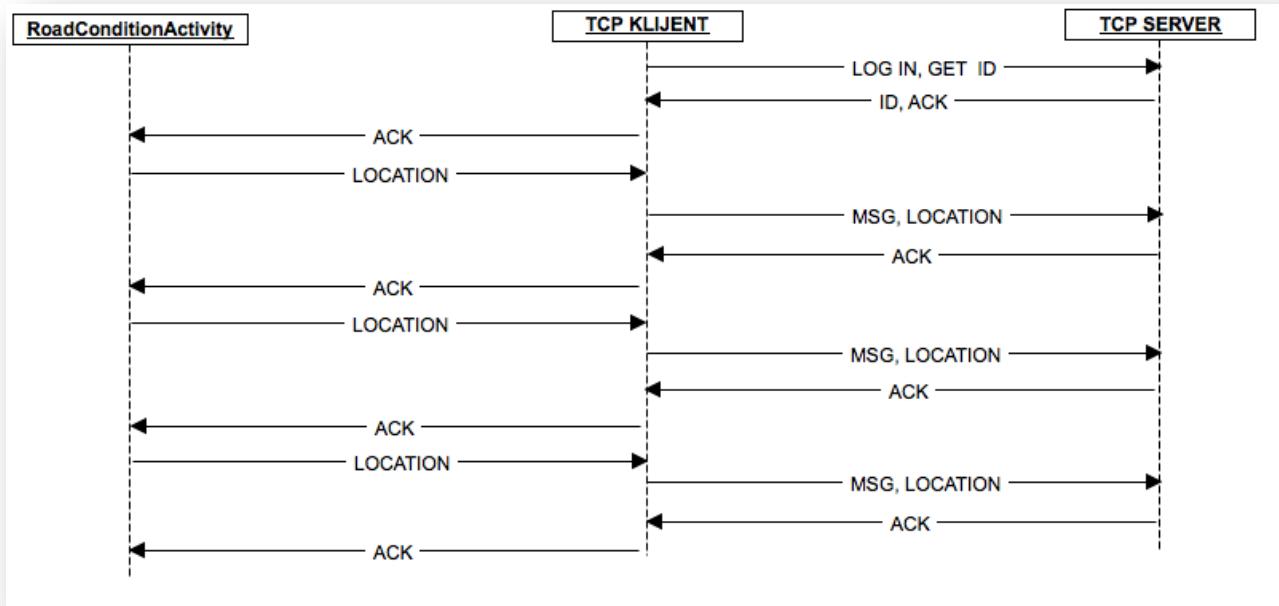
1. Automatska prijava trenutne lokacije korisnika,
2. Ručna prijava od strane korisnika nailaskom na karakterističan događaj,
3. Automatsko detektovanje neravnina putne infrastrukture,
4. Pozadinski alarm na prisustvo policijske patrole,
5. TCP komunikacija sa serverskom stranom sistema.



Slika 4. – Koncept rešenja

3.1 Automatska prijava lokacije

Programski modul koji se izvršava bez fizičke interakcije korisnika. Pokretanjem aplikacije, u rad se pušta GPS primopredajnik. Nakon uspešne detekcije satelita, na svaku promenu lokacije koja je očitana sa senzora, pokreće se TCP komunikacija sa serverskom stranom sistema (ukoliko postoji internet konekcija), kako bi server u svakom trenutku znao gde mu se nalazi korisnik.



Slika 5. – Scenario automatske prijave lokacije

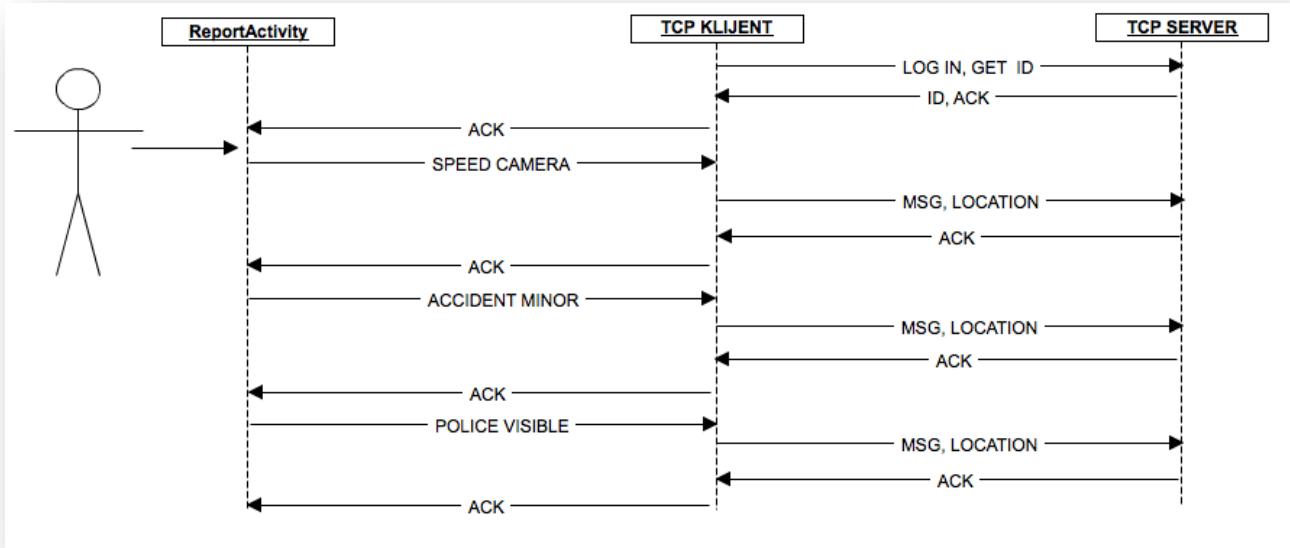
Ovaj pozadinski proces nema direktnog uticaja na korisnika, i samim tim ne ugrožava sigurnost vozača. Sve se izvršava takoreći ne primetno. Čak i ako veza sa satelitom nije uspostavljena, ili ne postoji internet konekcija, korisnik neće biti obavešten o tome, a ispunjenjem ovih uslova počinje/nastavlja (u zavisnosti da li su konekcije uspostavljene pa prekinute ili nisu ni uspostavljene iz nekog razloga) automatska prijava lokacije.

3.2 Ručna prijava događaja

Ručna prijava se odnosi na sve događaje koji su pod kontrolom korisnika. Na korisniku je da dojavi karakterističnu situaciju, i samim tim omogući ostalim korisnicima bezbedniju vožnju i podsticaj da učestvuju u korištenju aplikacije sa ciljem modernizacije saobraćaja i maksimalnim iskorištenjem modernih tehnologija našeg doba.

Startovanjem aplikacije potrebno je izabrati opciju za dojavu. Unutar svake od pet mogućih opcija, postoje pod opcije, koje unose veću preciznost pri dojavi. Nakon odabira događaja korisnik mora odabrati dugme za slanje informacije. Ukoliko ne postoji aktivna

internet veza, ili ukoliko GPS primopredajnik nije uspostavio vezu, korisnik se obaveštava upozorenjem, koje ga novodi da proveri konekcije i da pokuša ponovno slanje. Ukoliko su veze aktivne informacija prema serveru je poslata TCP protokolom, a korisnik o tome dobija obaveštenje.



Slika 6. – Scenario manuelne prijave događaja

3.3 Automatsko detektovanje neravnina

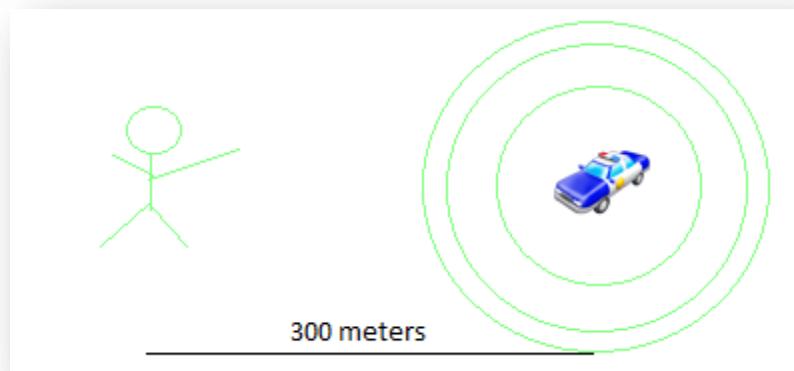
Analogno automatskoj prijavi lokacije, imamo prijavu detektovanih neravnina na određenom delu puta. Na početku se pokreće senzor za registrovanje pokreta uređaja (eng. *Accelerometer*). Ukoliko senzor registruje nagle promene vrednosti (promene položaja uređaja) automatski obaveštava serversku stranu o tome, opet putem TCP protokola.

Opet analogno automatskoj prijavi lokacije i ovaj modul se izvršava bez interakcije sa korisnikom. Da bi informacija bila korisna, potrebno je da uređaj na kojem je aplikacija pokrenuta, bude postavljen na predviđeno mesto za to. Jedino na taj način će se detektovati promena položaja koja je uslovljena samim vozilom.

Bitno za naglasiti je da pri realizaciji ovog programskog modula osetljivost senzora odnosno reakcija na intenzitet pomeraja je postavljena na takvu vrednost da je moguće registrirati samo neravnine koje svojim intenzitetom znatno utiču na bezbednost učesnika u saobraćaju kao i njihovih motornih vozila. U suprotnom bi server primao suviše informacije. Senzor uređaja pri mirovanju je u stanju detektovati i najmanje vibracije koje nastaju pri kretanju ostalih vozila.

3.4 Pozadinski alarm

Pri svakoj automatskoj prijavi lokacije, od strane servera očekuje se odgovor koji korisnika obaveštava da li je u blizini trenutne lokacije policijska patrola. Postoji mogućnost podešavanja udaljenosti na kojoj će korisnik blagovremeno biti obavešten o prisustvu policijske patrole. Na slici je prikazana udaljenost od 300m na kojoj će se generisati obaveštenje.



Slika 7. – Alarm na udaljenosti

Obaveštenje je u vidu pozadinskog alarma sa karakterističnim zvučnim efektom u trajanju od tri sekunde, koji se automatski pokreće pri dojavi servera. Prijem alarmne poruke od strane servera je prikazano na slici .

3.5 TCP komunikacija

Na klijentu je da otpočne komunikaciju sa serverom, sa definisanim parametrima konekcije. Na svaku poruku od strane klijenta, server je u obavezi da vrati odgovor o uspešnom prijemu poruke, kao i da je spreman na prijem naredne. Komunikacija počinje prijavljivanjem korisnika i zahtevanjem identifikacionog parametra, da bi korisnik u svakom trenutku znao sa kojim korisnikom razmenjuje informacije. Identifikacija se nalazi u odgovoru servera i od tog trenutka u svakoj narednoj poruci klijenta naveden je identifikacioni parametar. Iz odgovora servera se sazna da li je patrola u krugu korisnika, ili ne. Komunikacija se završava odjavljivanjem korisnika sa servera i otpuštanjem identifikacionog parametra, koji je sada raspoloživ za nekog narednog korisnika koji će se prijaviti u sistem.

U tabeli 1. su prikazani svi tipovi poruka koje se šalju od klijenta prema serveru. Takođe je dat kratak opis svake pojedinačne poruke.

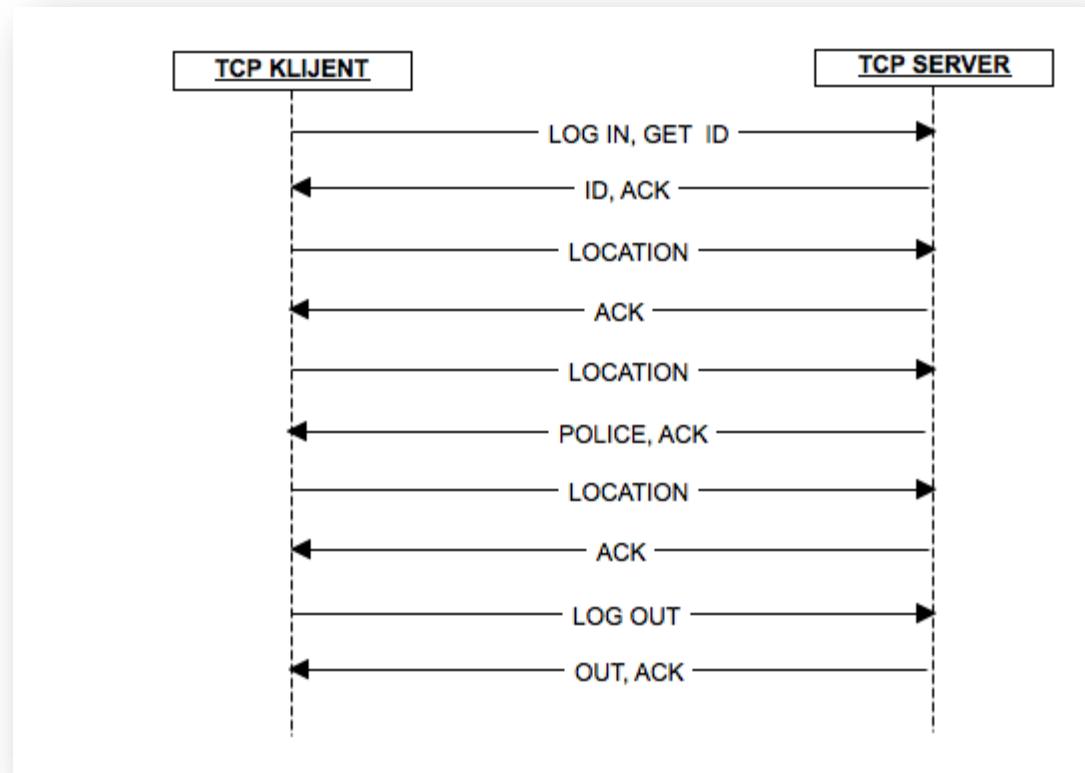
<u>Sadržaj poruke</u>	<u>Značenje poruke</u>
0/0/x kordinata/y kordinata	Automatsko javljanje lokacije korisnika – na svaku promenu
0/1/x kordinata/y kordinata	Prijavljivanje kod servera
0/2/x kordinata/y kordinata	Odjavljivanje sa servera
1/1/x kordinata/y kordinata	Policijska patrola - vidljiva
1/2/x kordinata/y kordinata	Policijska patrola – skrivena
2/1/x kordinata/y kordinata	Saobraćajna nesreća – manja šteta
2/2/x kordinata/y kordinata	Saobraćajna nesreća – veća šteta
3/1/x kordinata/y kordinata	Zagušenje – mali intenzitet
3/2/x kordinata/y kordinata	Zagušenje – srednji intenzitet
3/3/x kordinata/y kordinata	Zagušenje – totalni zastoj
4/1/x kordinata/y kordinata/komentar	Opasnost na putu
4/2/x kordinata/y kordinata/komentar	Opasnost pored puta
4/3/x kordinata/y kordinata/komentar	Opasnost – vremenske neprilike
5/1/x kordinata/y kordinata	Kamera – kontrola brzine
5/2/x kordinata/y kordinata	Kamera – kontrola na prolazak kroz crveno svetlo

Tabela 1. – Format poruka

Prilikom dojave o opasnostima koje mogu da ugroze bezbednost vozača, kao poslednji parametar dodat je komentar koji može da bude:

1. Object on road,
2. Construction,
3. Pothole,
4. Car stopped,
5. Missing sign,
6. Fog,
7. Hail,
8. Heavy Rain,
9. Snow,
10. Icy Road.

Na slici je prikazana komunikacija klijenta i servera, od početne prijave, prijema identifikacionog parametra pa sve do generisanja alarmne poruke i na kraju odjave korisnika sa servera.



Slika 8. – Scenario uspešnog prijema alarma

4. Programsко рење

Realizacija projektnog zadatka pisana je u *Eclipse Helios* razvojnom okruženju sa *ADT Plugin-om* i *Android SDK Tools-om*. Programski jezik koji je korišten pri realizaciji projekta je Java.

Rešenje je zahtevalo korišćenje ugrađenih sistemskih Android klasa, nasleđivanje, izmenu postojećih klasa i metoda, kao i dodavanje potpuno novih. Klase projektnog zadatka, kao i kraći opis, prikazane su u tabeli 2.

Sadrzaj paketa – klase sistema	Opis klase sistema
RoadConditionActivity.java	Klase za pokretanje glavne aktivnosti – početni izgled aplikacije za interakciju sa korisnikom. Pomoću globalnog pozicionog sistema prati se trenutna lokacija korisnika
ReportActivity.java	Klase zadužena za aktivnost čiji je sadržaj upravo glavni smisao aplikacije – sve opcije koje korisnik može da dojavi server
MenuActivity.java	Aktivnost koja sadrži određena podešavanja aplikacije, gašenje aplikacije
AccidentActivity.java	Aktivnost čiji je zadatak dojavljivanje serveru saobraćajnog udesa, ukoliko ga je korisnik uočio u toku vožnje
CameraActivity.java	Klase koja vodi brigu o dojavi postavljene kamere čiji je zadatak nadzor vozača u saobraćaju
HazardActivity.java	Dojava korisnika o opasnosti na putu (od radova na putu pa sve do loših vremenskih uslova)
JamActivity.java	Ukoliko korisnik smatra da je došlo do zagušenja na određenoj saobraćajnici može da dojavi serveru pomoću ove aktivnosti
PoliceActivity.java	Umesto “blendovanja” korisnik je u mogućnosti da dojavi serveru prisustvo policijske patrole na putu

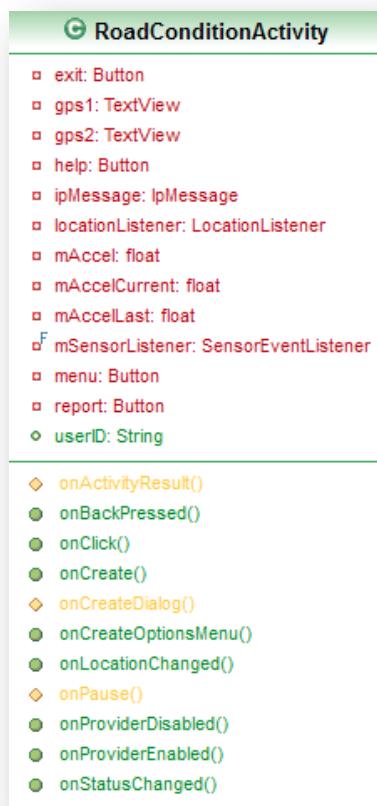
IpMessage.java	Klasa čiji je zadatak da sve dojave, od strane korisnika prema serveru i od servera prema korisniku, dojavi posredstvom tcp/ip protokola
----------------	--

Tabela 2. – Prikaz klase projektnog zadatka

Detaljnija specifikacija svake pojedinačne klase sledi u nastavku.

4.1 RoadConditionActivity klasa

Na slici 9. se može videti RoadConditionActivity klasa sa svim svojim metodama i atributima.

*Slika 9. – Prikaz klase RoadConditionActivity*

RoadConditionActivity klasa se prva instancira i ona nasleđuje sistemsku klasu androidia: Activity. Ona stranjuje početni izgled aplikacije sa četiri dugmeta, od kojih svaki ima svoju funkcionalnost, kao i pozadinsku sliku aplikacije. U nasleđenoj metodi onCreate se realizuje prethodno navedeno, kao i mogućnost da korisnik dalje odabere aktivnost, pritiskom na jedan od sledećih dugmadi:

- REPORT – pokreće se sledeća aktivnost u kojoj se nalazi celokupna interakcija sa serverom,

- MENU – aktivnost u kojoj se nalaze pojedine opcije kao što gašenje aplikacije, pojedina podešavanja, kao i reklama aplikacije,
- HELP – informacije o autoru i verziji aplikacije,
- EXIT – odjavljivanje sa servera a zatim gašenje aplikacije.

a sve to pomoću androidovog sistemskog interfejsa OnClickListener koji je implementiran u ovoj metodi.

U klasi je takođe realizovano konstantno ažuriranje trenutne lokacije korisnika, što omogućava implementiran sistemski interfejs LocationListener.

Neprekidno se prate vrednosti koje senzor uređaja očitava. Na svaku promenu vrednosti koje je senzor očitao, u pozadini se šalje informacija do servera, kako bi server u svakom trenutku znao gde mu se nalazi korisnik, a samim tim i da bi ga blagovremeno obavestio o situacijama na putu. Ova akcija se odvija nevidljivo za korisnika, automatski u pozadini. U slučaju da server primeti da bi korisnika morao obavestiti o nečem bitnom, generiše se pozadinski alarm koji upozorava korisnika(u našem slučaju vozača motornog vozila).

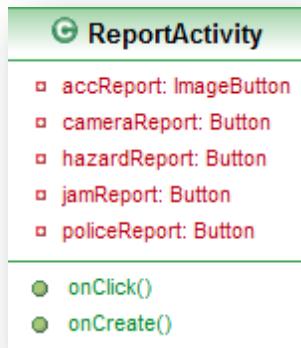
Rezultat RoadConditionActivity klase je početni ekran koji je dat na slici 10.



Slika 10. – Početni ekran aplikacije

4.2 ReportActivity klasa

Na slici 11. se može videti ReportActivity klasa sa svim svojim metodama i atributima.



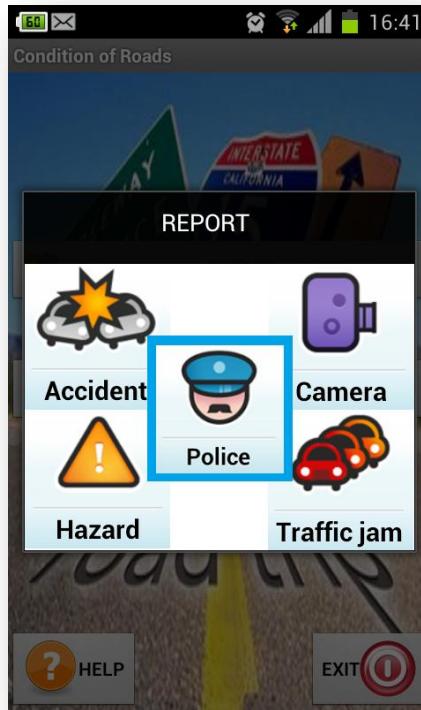
Slika 11. – Prikaz klase ReportActivity

ReportActivity klasa nasleđuje sistemsku klasu Activity. Startuje se izgled aktivnosti koja sadrži celokupnu interakciju korisnika sa serverom. Implementacijom interfejsa OnClickListener, na korisniku je da odabere jednu od šest aktivnosti:

- Accident – pokreće se sledeća aktivnost u kojoj se nalazi dojava serveru da se desila saobraćajna nesreća na određenoj lokaciji,
- Camera – aktivnost u kojoj se server dojavljuje postojanje kamere koja na određeni način prati tok saobraćaja, bila u pitanju brzina ili prolazak vozila na crveno svetlo semafora,
- Hazard – dojava serveru o mogućim opasnostima po učesnike saobraćaja, bili na putu ili kraj, ili pak lošim vremenskim uslovima na određenom delu deonice,
- Traffic Jam – aktivnost u kojoj se odabira nivo zagušenja u saobraćaju na trenutnoj lokaciji vozila,
- Police – dojavljivanje serveru o postojanju policijske patrole na trenutnoj lokaciji.

Sve prethodno navedeno realizovano je u metodi onCreate uz implementaciju OnClickListener-a. Bilo koji izbor korisnika će pokrenuti narednu aktivnost koja sadrži odabir malo preciznijih informacija o događaju na putu pre konačne dojave serverskom uređaju.

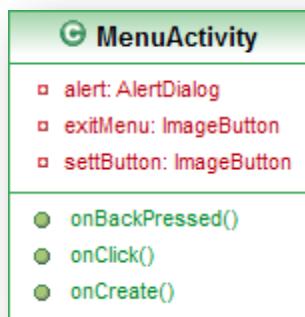
Instanciranjem objekta ove klase dobija se izgled koji je prikazan na slici 12.



Slika 12. – Izgled REPORT aktivnosti

4.3 MenuActivity klasa

Instanciranjem klase, koja je prikazana na slici 13., dobijamo aktivnost čiji je izgled jednostavan sa 3 dugmeta.



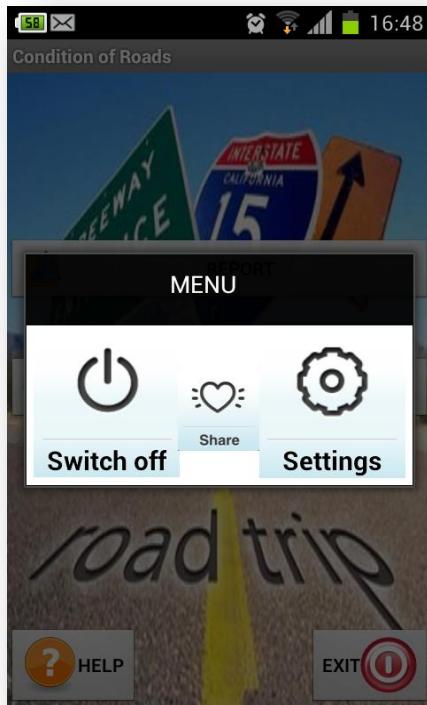
Slika 13. – Prikaz klase MenuActivity

I ova klasa nasleđuje sistemsku klasu Activity. Implementacijom interfejsa OnClickListener, korisniku su na raspolaganju:

- Switch off – gašenje aplikacije, kojem prethodi odjavljivanje sa servera. Takodje se gase svi potrebni senzori za rad aplikacije,
- Share – reklama aplikacije na stranici facebook.com ili na nekim drugim društvenim mrežama,
- Settings – podešavanja aplikacije, koja ne utiču direktno na funkcionalnost.

Pritiskom na dugme za gašenje aplikacije pokreće se novi prozor, u kojem se korisniku postavlja pitanje da li je siguran da želi da ugasi aplikaciju, a pored pitanja tu su i dva dugmeta, od kojih je jedan za potvrdu gašenja, a drugi za odustajanje.

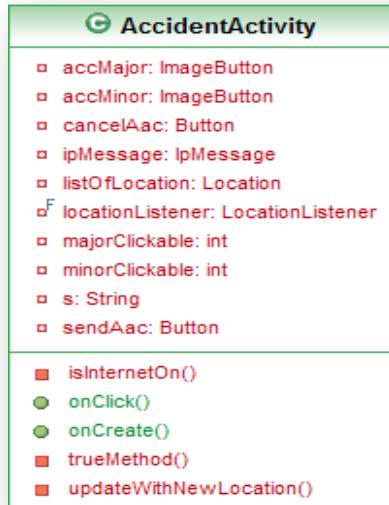
Instanciranjem objekta ove klase dobija se izgled koji je prikazan na slici 14.



Slika 14. – Izgled MENU aktivnosti

4.4 AccidentActivity klasa

Na slici 15. prikazana je klasa AccidentActivity sa svim svojim metodama i parametrima. Posredstvom ove klase korisnik vrši dojavu saobraćajne nesreće koju je zapazio na putu. Na njemu je da izabere između dve opcije: nesreća sa manjom štetom ili nesreća sa većom štetom. Dok god ne izabere jednu od ove dve opcije nije u mogućnosti da izvrši dojavu serveru.



Slika 15. – Prikaz klase AccidentActivity

U aktivnosti klase postoje četiri dugmeta:

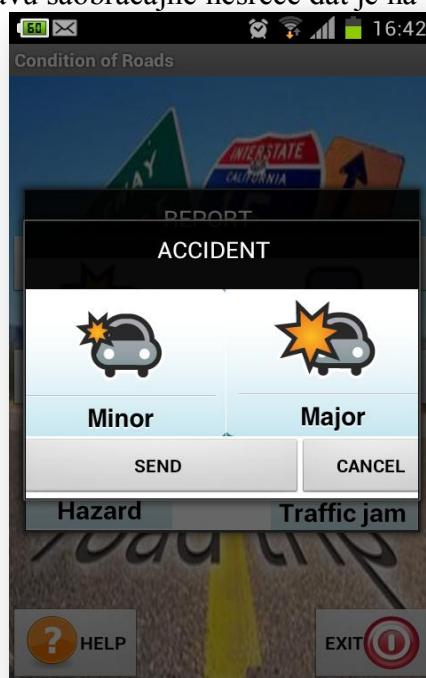
- Minor – dugme čiji odabir odgovara manjoj saobraćajnoj nesreći,
- Major – dugme čiji odabir odgovara većoj saobraćajnoj nesreći,
- SEND – nakon odabira jedne od prethodne dve opcije, posredstvom dugmeta SEND vrši se slanje informacija ka serveru,
- CANCEL – dugme za odustanak od bilo kog vida dojave, i povratak na REPORT aktivnost.

Osnovni izgled ove aktivnosti je obrađen u metodi onCreate uz implementaciju interfejsa OnClickListener koji dugmadi daje funkcionalnost.

Sva, osim dugmeta CANCEL, su međusobno sinhronizovana. Dugme za odustanak od dojave se može u bilo kom trenutku odabrati. Dojava se vrši u četiri koraka redom:

1. Selektovanje samo jednog od dva dugmeta koji opisuje nesreću. U datom trenutku samo jedno dugme može biti selektovano,
2. Pozadinska provera da li je mreža dostupna, da bi se podatci uspešno dostavili na cilj. Ovo omogućava metoda isInternetOn. Ukoliko mreža nije dostupna korisnik se obaveštava kratkom porukom da proveri dostupnost mreže,
3. Ako se uspostavi da mreža postoji, vrši se provera da li je moguće odrediti trenutnu lokaciju uređaja. Ukoliko lokaciju nije moguće odrediti ponovo se obaveštava korisnik da proveri dostupnost satelita,
4. Ukoliko su prva tri koraka ispoštovana, klikom na dugme SEND informacije su poslate serveru, a korisnik se vraća na aktivnost REPORT.

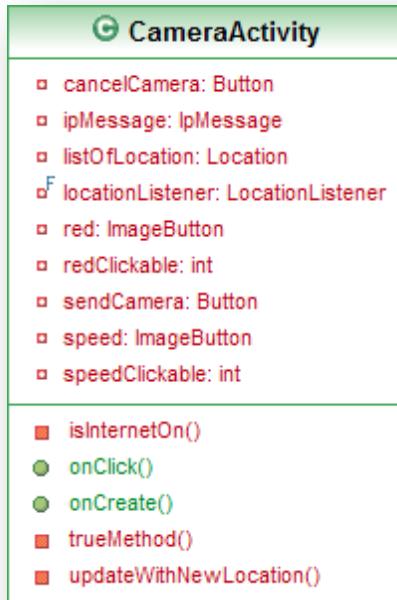
Izgled aktivnosti za dojavu saobraćajne nesreće dat je na slici 16.



Slika 16. – Izgled Accident aktivnosti

4.5 CameraActivity klasa

Na slici 17. prikazana je klasa CameraActivity sa svim svojim metodama i parametrima.



Slika 17. – Prikaz klase CameraActivity

Posredstvom ove klase korisnik vrši dojavu sa informacijom da je postavljena kamera koja prati stanje u saobraćaju. Ovim programski rešenjem pokrivena su dva slučaja saobraćajnih kamera. Prva od njih je kamera koja prati brzinu kretanja vozila odnosno nama poznatija pod imenom “radar”. Drugi vid kamere je ona koja prati prolazak vozila kroz crveno svetlo semafora. Na korisniku aplikacije je da izabere jednu od ove dve opcije i da dojavi serveru.

U aktivnosti klase postoje četiri dugmeta:

- Speed – dugme čiji odabir odgovara kameri za detekciju brzine vozila,
- Red light – dugme čiji odabir odgovara kameri namenjenoj za crveno svetlo,
- SEND – nakon odabira jedne od prethodne dve opcije, posredstvom dugmeta SEND vrši se slanje informacija ka serveru,
- CANCEL – dugme za odustanak od bilo kog vida dojave, i povratak na REPORT aktivnost.

Osnovni izgled je obraden u metodi onCreate uz implementaciju interfejsa OnClickListener koji dugmadi daje funkcionalnost.

Sva, osim dugmeta CANCEL, su međusobno sinhronizovana. Dugme za odustanak od dojave se može u bilo kom trenutku odabratи. Dojava se vrši u četiri koraka redom:

1. Selektovanje samo jednog od dva dugmeta koji opisuјe kameru. U datom trenutku samo jedno dugme može biti selektovano,
2. Pozadinska provera da li je mreža dostupna, da bi se podatci uspešno dostavili do servera. Ovo omogućava metoda isInternetOn. Ukoliko mreža nije dostupna korisnik se obaveštava kratkom porukom da proveri dostupnost mreže,
3. Ako se uspostavi da mreža postoji, vrši se provera da li je moguće odrediti trenutnu lokaciju uređaja. Ukoliko lokaciju nije moguće odrediti ponovo se obaveštava korisnik da proveri dostupnost satelita,
4. Ukoliko su prva tri koraka ispoštovana, klikom na dugme SEND informacije su poslate serveru, a korisnik se vraća na aktivnost REPORT.

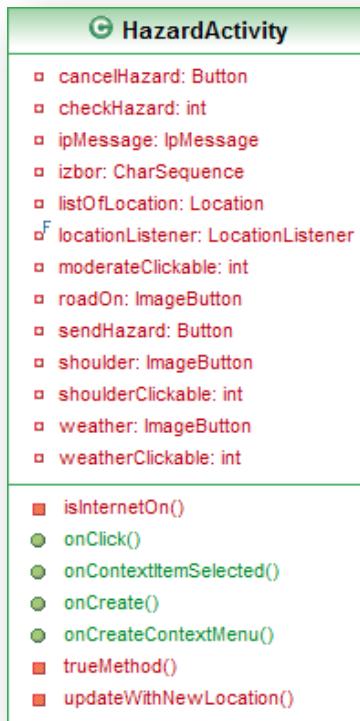
Izgled aktivnosti za dojavu postavljene kamere dat je na slici 18.



Slika 18. – Izgled Camera aktivnosti

4.6 HazardActivity klasa

Na slici 19. prikazana je klasa HazardActivity sa svim svojim metodama i parametrima.



Slika 19. – Prikaz klase HazardActivity

Pomoću ove aktivnosti korisnik obaveštava server o prisutnoj opasnosti na putu. Korisnik bira jednu od moguće tri opcije aplikacije. Svaka opcija ima nekoliko pod opcija da bi se što preciznije obavestili svi učesnici u saobraćaju o prisutnoj opasnosti. Prva opcija predstavlja opasnost prisutnu direktno na putu. Druga opcija sadrži skup opasnosti na koje se može naleteti kraj puta. Treća je upravo dojava o lošim vremenskim uslovima na deonici puta.

Prikaz opcija sa skupom pod opcija:

- On road – dugme čiji odabir odgovara opasnosti na putu,
 1. Object on road – predmet na putu koji izaziva opasnost,
 2. Construction – radovi na putu,
 3. Pothole – loše stanje puta, opasne neravnine koje izazivaju opasnost po učesnike i mogu naneti štetu vozilu.
- Shoulder – dugme čiji odabir odgovara opasnosti kraj puta,
 1. Car stopped – zaustavljeni vozilo kraj puta,
 2. Missing sign – nedostatak znaka može uticati na logiku učesnika, a samim tim i uzrokovati ozbiljnije probleme.

- Weather – loši vremenski uslovi na putu,
 1. Fog – magla kao jedan od naj ozbiljnijih vremenskih poteškoća,
 2. Hail – grad koji može prouzrokovati ozbiljna oštećenja vozila,
 3. Heavy Rain – olujni pljuskovi znatno smanjuju vidljivost,
 4. Snow – deonica sa snežnim nanosima,
 5. Icy Road – poledica kao predstavnik zimskih uslova vožnje.
- SEND – nakon odabira jedne od prethodnih opcija, posredstvom dugmeta SEND vrši se slanje informacija ka serveru,
- CANCEL – dugme za odustanak od bilo kog vida dojave, i povratak na REPORT aktivnost.

Dodatne pod opcije su realizovane kao skup dodatnih opcija u metodi onCreateContextMenu.

Osnovni izgled je obrađen u metodi onCreate uz implementaciju interfejsa OnClickListener koji dugmadi daje funkcionalnost.

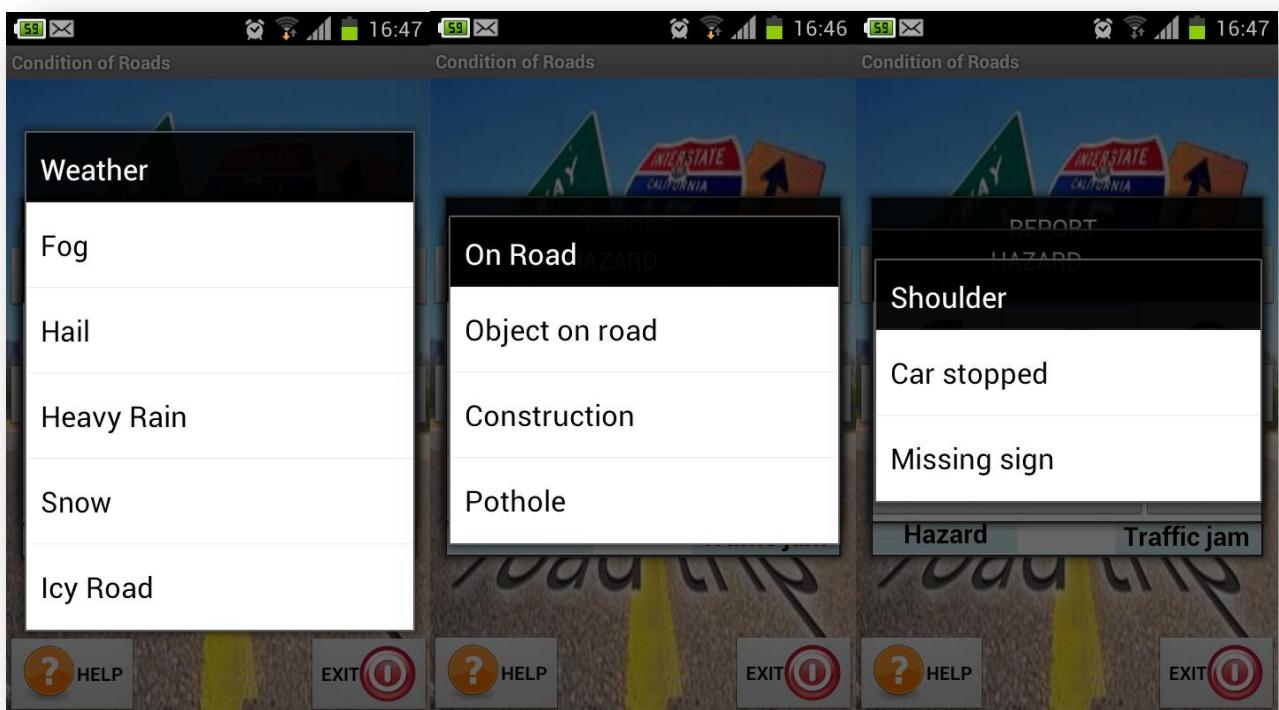
Sva, osim dugmeta CANCEL, su međusobno sinhronizovana. Dugme za odustanak od dojave se može u bilo kom trenutku odabrati. Dojava se vrši u četiri koraka redom:

1. Odabirom jednog od tri dugmeta otvara se dodatni meni sa listom preciznijih dodatnih opcija, radi što bolje dojave serverskoj strain sistemu. U datom trenutku moguće je samo jednu od ponuđenih opcija, odabrati,
2. Pozadinska provera da li je mreža dostupna, da bi se serverskoj strain uspešno dojavila opasnost. Ovo omogućava metoda isInternetOn. Ukoliko mreža nije dostupna korisnik se obaveštava kratkom porukom da proveri dostupnost mreže,
3. Ako se uspostavi da mreža postoji, vrši se provera da li je moguće odrediti trenutnu lokaciju uređaja. Ukoliko lokaciju nije moguće odrediti ponovo se obaveštava korisnik da proveri dostupnost satelita,
4. Ukoliko su prva tri koraka ispoštovana, klikom na dugme SEND informacije su poslane serveru, a korisnik se vraća na aktivnost REPORT.

Izgled aktivnosti za dojavu opasnosti na putu dat je na slici 20. dok na slici 21. je prikazan podskup opcija dugmeta Weather, On Road i Shoulder.



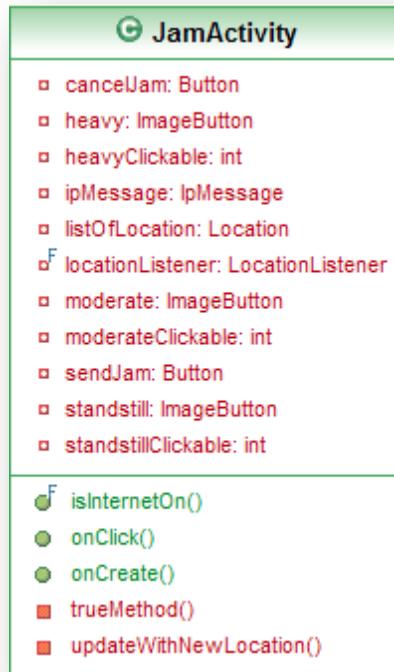
Slika 20. – Prikaz izgleda aktivnosti Hazard



Slika 21. – Opcije u meniju dugmeta Weather, On Road, Shoulder

4.7 JamActivity klasa

Na slici 22. prikazana je klasa JamActivity sa metodama i parametrima. Izvršavanjem ove klase korisnik vrši dojavu pojačanog saobraćaja na određenoj deonici puta, koji ostali učesnici mogu izbeći i samim tim smanjiti gužvu, kao i smanjiti sebi nepotrebno čekanje u koloni. Pošto zagušenje može da varira po intenzitetu, tako je i korisniku aplikacije dato na izbor tri nivoa zagušenja. Da bi uspešno dojavio serverskoj strani na njemu je da izabere nivo zagušenja.



Slika 22. – Prikaz klase JamActivity

U aktivnosti klase postoji pet dugmadi:

- Moderate – zagušenje minimalnog intenziteta,
- Heavy – dugme čiji odabir odgovara zagušenju srednjeg intenziteta,
- Standstill – potpuni zastoj u saobraćaju na određenom delu deonice,
- SEND – nakon odabira jedne od prethodna tri intenziteta, posredstvom ovog dugmeta vrši se slanje informacija ka serverskoj strani,
- CANCEL – dugme za odustanak od bilo kog vida dojave, i povratak na REPORT aktivnost.

Za izgled ove aktivnosti zadužena je onCreate metoda uz implementaciju interfejsa OnClickListener. Osim dugmeta za odustanak od dojave, ostala dugmad su sinhronizovana. Poruka serveru neće biti poslata ako prethodno nije selektovan samo jedan intenzitet zagušenja, proverena dostupnost mreže, kao i dostupnost satelita radi dobijanja trenutne

lokacije. Po uspešnom slanju poruke na dugme SEND korisnik se vraća na aktivnost REPORT.

Izgled aktivnosti za dojavu zagušenja u saobraćaju dat je na slici 23.



Slika 23. – Izgled Traffic jam aktivnosti

4.8 PoliceActivity klasa

U današnje vreme jako je teško odlučiti koja od realizovanih funkcija je najbitnija za korisnike. Kada je ljudska sigurnost ugrožena na bilo koji način, tada sve drugo postaje manje važno osim ljudskog života. Upravo iz tog razloga narednu opciju, koja je realizovana u ovom projektnom zadatku, nećemo nazvati najbitnijom već opcijom koja korisniku privlači najviše pažnje.

Posredstvom ove klase korisnik vrši dojavu sa informacijom da je prisutna policijska patrola na lokaciji iz poruke koja je poslata serverskoj strani. Ovim programskim rešenjem pokrivena su dva slučaja patrola:

- Visible – dugme čiji odabir odgovara patroli koja je lako uočljiva iz vozila,
- Hidden – dojava o patroli koja je teže vidljiva iz bilo kog razloga.

Pored ova dva dugmeta tu su i dugme za slanje poruke SEND i dugme za odustanak od dojave CANCEL. I u ovoj klasi za izgled aktivnosti zadužena je metoda onCreate, a za funkcionalnost dugmadi sistemski interfejs OnClickListener. i u ovom slučaju aktivnost dugmadi je sinhronizovana a da bi se uspešno prosledila dojava serveru, a samim tim i ostalim korisnicima aplikacije, potrebno je ispoštovati korake:

1. Selektovanje samo jednog od dva dugmeta za patrolu,
2. Pozadinska provera da li je mreža dostupna, da bi se podatci uspešno dostavili do servera. Ovo omogućava metoda isInternetOn. Ukoliko mreža nije dostupna korisnik se obaveštava kratkom porukom da proveri dostupnost mreže,
3. Ako se uspostavi da mreža postoji, vrši se provera da li je moguće odrediti trenutnu lokaciju uređaja. Ukoliko lokaciju nije moguće odrediti ponovo se obaveštava korisnik da proveri dostupnost satelita,
4. Ukoliko su prva tri koraka ispoštovana, klikom na dugme SEND informacije su poslate serveru, a korisnik se vraća na aktivnost REPORT.

Na slici 24. prikazana je klasa CameraActivity sa svim svojim metodama i parametrima.



Slika 24. – Prikaz klase PoliceActivity

Izgled aktivnosti za dojavu o policijskoj patroli dat je na slici 25.



Slika 25. – Izgled Police aktivnosti

4.9 IpMessage klasa

Na slici 26. prikazana je klasa IpMessage sa svim svojim metodama i parametrima.



Slika 26. – Prikaz klase IpMessage

Pri svakoj dojavi korisnika instancira se objekat upravo ove klase koja sadrži metodu runTcpClient.

Na samom početku vrši se priprema odlazne poruke. Kodovanje poruke prikazano je u tabeli u poglavlju koncept rešenja. Pored svih dojava koje korisnik obavlja ručno,instanciranje ove klase se vrši i pri pokretanju aplikacije radi prijave serveru, zatim pri zatvaranju aplikacije radi odjavljivanja sa servera da ne bi došlo do preopterećenja servera.

Pored ovih slučajeva, metoda runTcpClient se poziva na svaku promenu lokacije, da bi se server obavestio o kretanju korisnika, kao i pri zapažanju većih neravnina na putu, koje se očitavaju sa senzora pokreta (eng. *Accelerometer*).

Na svaku odlaznu poruku, čeka se odgovor servera jer se koristi tcp protokol pri komunikaciji. Dolazna poruka se obrađuje. Ako je u pitanju format poruke koji javlja prisustvo policijske patrole, na određenoj udaljenosti od korisnika (udaljenost pri kojoj server javlja prisustvo patrole određuje direktno server) generiše se alarm, čiji je zadatak da blagovremeno obavesti korisnika (vozača) na prisustvo patrole.

5. Rezultati

Ispitivanje rada projektnog sistema obavljeno je nizom testiranja.

Testiranja su vršena nad nekoliko različitih uređaja sa android operativnim sistemom. Pokazalo se, da se uređaji ne ponašaju na isti način pri istim situacijama (kvalitet i preciznost određivanja lokacije GPS primopredajnika varira od uređaja do uređaja). Ustanovljeno je da prilikom pozicioniranja na određnoj lokaciji u stanju mirovanja, GPS unosi određeni nivo greške i počinje "razbacivati" lokaciju oko sebe. Prilikom kretanja uređaja u vozilu "razbacivanje" ne postoji ili je veoma malo (u prihvatljivim granicama). Greška tokom kretanja varira i u većini slučajeva je manja od greške prilikom mirovanja.

Prilikom testiranja komunikacije u drugom smeru, tj prijema alarmne poruke, na određenoj udaljenosti od unapred izgenerisane poruke policijske patrole, alarm se uspešno pokrenuo i obavestio korisnika (vozača) o postojanju patrole. Za jedan tipičan događaj alarm se generiše samo jednom da bi obavestio korisnika na prisustvo patrole. Kada se napusti zona ove patrole, nailaskom na drugu, alarm se ponovno generiše.

6. Zaključak

Zadatak rada bio je sistem koji bi posredstvom servera umrežio vozila, i dostavio učesnicima u saobraćaju neophodne informacije.

Obzirom da je pokriveno samo nekoliko najbitnijih opcija za korisnika, ovaj sistem je moguće proširiti kako u vidu dodatne funkcionalnosti i pokrivanja šireg skupa događaja, tako i na plasiranju aplikacije potencijalnim korisnicima, kako bi se modernizovao saobraćajni sistem i popravio standard današnjih vozača.

Samo neki od mogućih skupa nadgledanih događaja:

1. Tačke od interesa (benzinske stanice, slobodan parking prostor, restorani i ostali ugostiteljski objekti),
2. Praćenje motorista koji su danas najugroženija vrsta u saobraćaju,
3. Mogućnost da korisnik kada primeti karakterističan slučaj, ima mogućnost ukoliko želi da izvrši dojavu fotografiom, ili pak unese neki dodatni komentar (unos preko tastature ili unos glasovnim komandama),
4. Vozila koja se kreću deonicama poput auto-puta sa većom brzinom kretanja mogla bi međusobno komunicirati, razmenjivati informacije, koje bi na vreme upozorile vozača na neki vid opasnosti.
5. Omogućiti svakom pojedinačnom korisniku kreiranja naloga na serverskoj strani, kako bi mogao unapred definisati putanju kretanja, da bi dobio samo informacije od značaja za njega,
6. Postojanje liste događaja koja se automatski osvežava na određeni vremenski interval, tako da bi korisnik imao pregled svih informacija u bilo kom trenutku.

Ovo su samo neke od stvari koje bi bilo jako korisno implementirati u sistem.

7. Literatura

- [1] Wikipedia: *The Free Encyclopedia* , <http://www.wikipedia.org>
- [2] Android Developers: <http://developer.android.com>
- [3] Ed Burnette: *Hello, Android: Introducing Google's Mobile Development Platform*, North Carolina, 2008
- [4] Donn Felker, John Wiley & Sons: *Android Application Development For Dummies*, USA, 2010
- [5] Reto Meier: *Professional Android Application Development*, USA, 2008