

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Becele Miha

Prikaz senzorskih meritev z aplikacijo Android

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mira Trebar

Ljubljana, 2013

Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljane ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.



Št. naloge: 00357/2012

Datum: 04.12.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **MIHA BECELE**

Naslov: **PRIKAZ SENZORSKIH MERITEV Z APLIKACIJO ANDROID**
PRESENTATION OF SENSOR MEASUREMENTS WITH ANDROID
APPLICATION

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje


Tematika naloge:

Kandidat naj v diplomski nalogi izdela mobilno aplikacijo za prikaz senzorskih meritev na mobilnem-pametnem telefonu z operacijskim sistemom Android. Podatki o temperaturah, ki so bili zbrani v postopku sledljivosti živil z uporabo tehnologije RFID v prehranski verigi se nahajajo na strežniku. Zapis je definiran z razširljivim označevalnim jezikom xml. Aplikacija naj omogoča izbiro datoteke in prikaz rezultatov hladne verige tako kot seznam temperatur s časovno oznako in predstavitev v grafu s podanim območjem priporočljivih temperatur. Testiranje naj poteka z uporabo realnih podatkov na strežniku Laboratorija za računalniško arhitekturo.

Mentor:


doc. dr. Mira Trebar

Dekan:


prof. dr. Nikolaj Zimic



IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani Miha Becele, z vpisno številko 63070226, sem avtor diplomskega dela z naslovom:

Prikaz senzorskih meritev z aplikacijo Android.

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal/-a samostojno pod mentorstvom doc. dr Mire Trebar,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne _____ Podpis avtorja: _____

ZAHVALA

Za pomoč pri izdelavi diplomskega dela se zahvaljujem vsem, ki so mi kakorkoli pripomogli k nastanku dela, še posebej mentorici doc. dr. Miri Trebar.

Staršem.

KAZALO

Povzetek	1
Abstract	2
1 Uvod.....	3
2 Mobilne tehnologije in programska oprema.....	4
2.1 Razvoj mobilnih omrežij	4
2.1.1 Mobilne naprave	6
2.2 Mobilne platforme.....	8
2.2.1 Apple - iOS	9
2.2.2 Microsoft - Windows Mobile.....	10
2.2.3 Research In Motion - BlackBerry	11
2.2.4 Symbian Ltd. – Symbian OS	11
2.2.5 Google - Android.....	12
2.3 Eclipse	17
2.4 Programska orodja	18
3 Razvoj mobilne aplikacije	19
3.1 Opis senzorskih meritev	19
3.2 Načrt aplikacije	21
3.3 Pridobitev in prikaz podatkov.....	24
3.4 Izdelava logotipa in grafik za gumbe	28
4 Test aplikacije	30
4.1 Testni primeri	30
4.2 Android Emulator	30
4.3 Testni mobilnik	31
4.4 Testiranje aplikacije	33
4.5 Namestitev aplikacije	35
5 Zaključek	36

Seznam uporabljenih kratic in simbolov

API	Programski vmesnik (ang. Application Programming Interface)
ADT	Razvojno orodje za Android (ang. Android Development Tools)
AVD	Android virtualna naprava (ang. Android Virtual Device)
EDGE	Nadgradnja sistema GSM (ang. Enhanced Data rates for GSM Evolution)
GPRS	Paketno orientirano mobilno omrežje (ang. General Packet Radio Service)
GPS	Sistem globalnega določanja lege (ang. Global Positioning System)
GSM	Mobilno omrežje (ang. Global System for Mobile Communications)
HSDPA	Hitrejši protokol v omrežju UMTS (ang. High-speed circuit-switched data)
HSPA	Skupno ime za HSDPA in HSUPA (ang. High-Speed Packet Access)
HTML	Jezik za označevanje nadbisedila (ang. Hyper Text Markup Language)
IMAP	Internetni protokol na aplikacijskem sloju (ang. Internet Message Access Protocol)
LTE	Standard mobilnega omrežja (ang. Long Term Evolution)
MMS	Multimedijski sporočilni sistem (ang. Multimedia Messaging System)
MPEG	Komisija za razvoj video in audio kodiranja (ang. Moving Picture Experts Group)
NMT	Analogno mobilno omrežje (ang. Nordic Mobile Telephone)
OHA	Skupina 84 tehnologij in mobilnih podjetij (ang. Open Handset Alliance)
OS	Operacijski sistem (ang. Operating System)
POP3	Internetni protokol na aplikacijskem sloju (ang. Post Office Protocol version 3)
SDK	Nabor orodij za razvoj programske opreme (ang. Software Development Kit)
SMS	Kratka tekstovna sporočila (ang. Short Message Service)
SMTP	Protokol za prenos elektronske pošte (ang. Simple Mail Transfer Protocol)
TCP/IP	Internetni sklad protokolov (ang. Transmission Control Protocol/Internet Protocol)
UMTS	Mobilno omrežje tretje generacije (ang. Universal Mobile Telecommunications System)
USB	Univerzalno serijsko vodilo (ang. Universal Serial Bus)
VoIP	Paketni prenos zvoka po omrežju (ang. Voice over IP)
WiMAX	Standard za brezžično komunikacijo (ang. Worldwide Interoperability for Microwave Access)
XML	Označevalni jezik (ang. Extensible Markup Language)

Povzetek

Pametni mobilni telefoni, so v zadnjih letih doživeli razcvet na področju mobilne telefonije. Googlov operacijski sistem Android iz dneva v dan pridobiva na popularnosti in prevzema vlogo vodilnega operacijskega sistema za mobilne naprave. Cilj diplomskega dela je izdelava mobilne aplikacije za prikaz senzorskih meritev. Ti podatki so temperature in shranjeni na spletnem strežniku v obliki XML. Pregledujemo jih na pametnem telefonu, ki ima nameščen Android mobilno platformo. Za lažje razumevanje celotne vsebine je v prvem delu opisana mobilna tehnologija in programska oprema, vse od mobilnih omrežij, mobilnih naprav, pametnih telefonov in predstavitev petih vodilnih mobilnih platform. V drugem delu je podan načrt mobilne aplikacije in njene funkcije. Predstavljeni so podatki, katere prebiram iz strežnika. Na voljo imamo dva načina prikaza podatkov, tako tekstovni način, kjer se nam za vsako meritev prikaže izmerjena temperatura skupaj s časom in datumom nastanka meritve. V grafičnem načinu pa izrišemo graf, kjer vrednosti temperatur predstavimo na vertikalni osi odvisne od datuma meritve, ki so na horizontalni osi. Na koncu predstavljamo tudi testiranje aplikacije z testnimi primeri.

Ključne besede: Android, pametni telefon, temperatura, omrežje.

Abstract

Smartphones have bloomed in the market of mobile devices in the past years. Google's operating system Android is daily gaining on popularity and is overtaking the role of the primary operating system for mobile devices. The goal of my thesis is a development mobile application for showing measurements of different sensors. This is data, saved on a server in an XML form, which can be viewed on a smartphone with an Android platform installed. To understand the contents of the thesis I have described the mobile technology and software, going from the mobile networks, mobile devices, smartphones to presenting the five biggest mobile platforms. In the second part of the thesis I present the structure of the mobile application in it's functionalities. I present the data, which is read from the server. There are two available views to show the data, one is a text based representation, where we see the temperature along with the time and date. The other is a graphical representation, where a graph is drawn, showing the values of the temperatures on a vertical axis dependent of the date of the measurements on the horizontal axis. In the end there is a brief representation of the application testing.

Keywords: Android, smartphone, temperature, network.

1 Uvod

Danes smo v času, ko je mobilni telefon nuja, ko ima ogromno ljudi v lasti pametne telefone. Samo v lanskem letu je bilo prodanih (po podatkih IDC) skoraj 500 milijonov takšnih naprav, delež teh telefonov pa se iz dneva v dan bliskovito povečuje, saj v nekaterih državah delež teh naprav dosega že skoraj 50% (Velika Britanija, ZDA, Kanada ...)[1]. Med temi podatki velja izpostaviti veliko rast nakupov, saj se ob vsakem, novem letu prodaja poveča za 30% in več (odvisno od države).

Posledično se s prodajo naprav povečuje tudi izdelovanje in uporaba aplikacij. Samo Android aplikacij je od marca 2009 do junija 2012 nastalo več kot pol milijona (objavljenih na Google Play), število prenosov pa se je povzpelo na 20 milijonov [2].

Prav zaradi teh podatkov, zaradi ogromnega zanimanja za pametne telefone in zaradi ogromnega števila prenosov aplikacij smo se odločili, da v diplomskem delu razvijemo aplikacijo za mobilne telefone.

Zakaj ravno Android? Android je odprtokodni operacijski sistem, ki je po mojih izkušnjah preprost za uporabo in nam dovoljuje veliko opcij in rešitev, to pa je bil glavni razlog da smo izbrali Android mobilno platformo. Prav tako nam programska oprema (razvija jo podjetje Google z Open Hand Set Alliance) omogoča razvoj aplikacij za mobilne naprave. Delež Android naprav je več kot 60% (po podatkih IDC in Garthner [1]), sledi mu iOS z nekaj več kot 20%.

V prvem delu diplomske naloge predstavim tehnologije in programsko opremo, ter mobilne tehnologije, vse od prve generacije, pa do današnje četrte generacije, ki je trenutno še v času razvijanja. Predstavimo pet mobilnih platform, ki so trenutno vodilna v svetu, to so iOS, Windows Mobile, BlackBerry, Symbian in Android za katero je bil razvit program, ter razvojno okolje Eclipse in programska orodja.

V drugem delu diplomske naloge je predstavljena aplikacija z namenom prikaza senzorskih meritev, ki bi imela informativni značaj. Z aplikacijo pridobimo informacije o izmerjenih temperaturah, ki so bile prebrane s čitalniki in shranjena na spletnem strežniku. V tretjem poglavju predstavim razvoj mobilne aplikacije, vse od opisa zahtev do načrta aplikacije in opisa posameznega okna. Četrto poglavje je namenjeno testiranju aplikacije. Najprej predstavimo HTC mobilni telefon, na katerem smo aplikacijo testirali, nato podamo različne primere uporabe same aplikacije in predstavimo teste, s katerimi smo aplikacijo testirali. Na koncu podam zaključek in literaturo.

2 Mobilne tehnologije in programska oprema

Izraz »mobilne tehnologije« ne gre jemati dobesedno, saj tehnologije same po sebi niso mobilne. Mobilni smo uporabniki, ki tehnologije uporabljamo. Tehnologije so pravzaprav prenosljive, vendar jih imenujemo mobilne, ker so prirejene za mobilno uporabo. To pomeni, da so majhne, lahke in priročne, da delujejo na baterije in so enostavne za uporabo. Z izrazom mobilne tehnologije ponavadi opisujemo tehnologije, ki jih uporabljamo, ko smo na poti, in jih v grobem razdelimo na mobilna oziroma brezžična omrežja, mobilne naprave, mobilne platforme in programska orodja.

Mobilne tehnologije bodo v kratkem postale center človeške evolucije, saj ukrivljajo dimenzijo časa in prostora in kot takšne vplivajo na vse sfere našega življenja. Razvoj gre v smeri najrazličnejših scenarijev uporabe - od upravljanja gospodinjskih aparatov, plačevanja položnic, nakupa vstopnic za nogometne tekme do navezovanja stikov preko socialnih omrežij, navigiranja z uporabo zemljevidov in neposrednega sporočanja širom po svetu.

2. 1 Razvoj mobilnih omrežij

Mobilne tehnologije se iz leta v leto hitreje razvijajo in so med uporabniki vedno bolj razširjene. Razlogi za hitro razvijanje mobilnih tehnologij so huda konkurenčnost izdelovalcev opreme in ponudnikov storitev ter veliko število uporabnikov, ki so pripravljeni plačevati za nove izdelke in storitve. V nadaljevanju bomo opisali razvoj mobilnih omrežij skozi štiri generacije.

Prva generacija (1G)

Omrežje prve generacije je zgradilo podjetje NTT leta 1979 na Japonskem, ki je pokrivalo območje mesta Tokio. Dve leti pozneje se je pojavilo omrežje NMT (Nordic Mobile Telephone), ki smo ga konec 80.tih let spoznali tudi pri nas z ustanovitvijo družbe Mobitel. NTT sistem, ki je pokrival celotno Japonsko, se šteje kot prvo, ki je pokrilo celotno območje države. Prvi sistem, ki je omogočal prehode iz različnih držav, t.i. gostujoči klici je bil sistem NMT, saj so bile prvotno podpornice tega omrežja Danska, Norveška, Švedska in Finska. Tehnično gledano se vsi sistemi prve generacije razlikujejo od prejšnjih sistemov po tem, da so bili ti sistemi zmožni »nevidno« prenesti klic iz ene celice v drugo tako, da uporabnik tega sploh ni opazil. Govor sam pa se je še vedno prenašal kot analogni signal, v nekaterih omrežjih je bil le moduliran z višjo frekvenco [3].

Druga generacija (2G)

Kratiko GSM enačimo z drugo generacijo, vendar obstaja več tehnologij. GSM je bil prvi, saj je že leta 1991 prvo omrežje postavil Finski telekomunikacijski operater. Z drugo generacijo je bilo uporabnikov vedno več, to pa je pripomoglo tudi k temu, da so se telefoni drastično zmanjšali – iz nekaj kilogramov na nekaj 100 gramov, kar je za povprečnega uporabnika prinašalo veliko prednost. V drugi generaciji je bil digitaliziran in kriptiran govor, kar je bila največja razlika v primerjavi z prvo generacijo in tako je bila zagotovljena večja zasebnost kot v prejšnjih omrežjih. Z drugo generacijo so se vpeljala danes tako priljubljena kratka sporočila, ki so bila v začetku drugotnega pomena, vendar je nepričakovano kmalu postala zelo popularna storitev. Skozi leta so GSM omrežja nadgrajevali, tako da so vmes nastala še t.i. 2.5G omrežja, ki so vključevala poleg govorne komunikacije tudi podatkovni dostop do spleta. Prva taka nadgradnja je bil GPRS, ki je uvedel obračunavanje po dejansko prenesenih paketih ter višje hitrosti kot klasični klicni dostop do spleta. EDGE je nadgradnja GPRSja, ki je omogočal višje hitrosti podatkovnih prenosov ter se v literaturi omenja kot 2.75G oz. kot zadnji mejnik prehoda v 3G omrežja [4].

Tretja generacija (3G)

Še pred pojavom GPRS in EDGE sistemov je konzorcij GSM izglasoval nov standard UMTS, ki tehnično gledano pokriva več različnih standardov implementacije, v osnovi pa obljublja višje hitrosti podatkovnih prenosov. Še preden pa se je nova generacija uveljavila, sta se pojavili že zgoraj omenjeni nadgradnji GSM omrežja GPRS in EDGE, ki omogočata hitrosti prenosa podatkov blizu klasifikaciji 3G omrežij. HSDPA je danes najbolj množično uporabljen standard za prenos podatkov, ki zahteva uporabo UMTS omrežja. HSDPA+ je trenutno najhitrejša realizacija tega standarda in omogoča prenose podatkov do 84Mbit/s. Tehnično se 3G omrežja ne razlikujejo veliko od 2G omrežij, omogočajo pa višje prenosne hitrosti [5].

Četrta generacija (4G)

Najnovejša generacija mobilnih omrežij se je pojavila leta 2009 s pojavom dveh različnih tehnologij: WiMAX in LTE. Potreba po novi generaciji mobilnega omrežja se je pojavila s povečanjem števila t.i. pametnih telefonov, ki so spremenili navade uporabnikov mobilnih omrežij – vedno bolj se namreč naprave uporabljajo za prenos podatkov in vedno manj le za glasovno sporočanje. Trenutno je večina WiMAX in LTE omrežij v testni fazi, saj na trgu še ni dovolj naprav, ki bi izkoriščala prednosti teh omrežij. Tehnično se ta generacija razlikuje od prejšnjih v tem, da je govor preprosto samo eden izmed množice podatkov, ki se prenašajo po omrežju. Tako se govor prenaša preko VoIP protokolov, kar v veliki večini pomeni prenos podatkov preko TCP/IP protokola [6].

2.1.1 Mobilne naprave

Mobilne naprave so elektronske naprave, ki jih uporabljamo za brezžično komunikacijo z drugimi uporabniki ali za povezovanje prek mobilnih omrežij do storitev na internetu. Prve mobilne naprave so bili telefoni, ki so bili izjemno veliki in so omogočali le telefoniranje. Z razvojem sodobnih računalniških in informacijsko komunikacijskih tehnologij so mobilni telefoni postali bolj priročni, manjši in komunikacijsko veliko bolj zmogljivi. Omogočajo več različnih oblik komunikacije (govor, besedilo, zvok in video), boljšo kakovost storitev in hitrejši prenos podatkov.

Za brezžično komunikacijo ne uporabljamo samo mobilne telefone, ampak so trenutno v uporabi zelo različne mobilne naprave (slika 1), ki jih glede na obliko in vrsto ter način uporabe razdelimo na:

- prenosne računalnike,
- tablične računalnike,
- ročne računalnike,
- dlančnike,
- mobilne telefone, ali t. i. pametne telefone.

Mobilne naprave nam omogočajo sporočanje, različne informacijske storitve in predvajanje večpredstavnostnih vsebin, kot so večpredstavnostna mobilna sporočila MMS (Multimedia Message Service), mobilni dostop do interneta, spletna pošta, glasba in informacijske storitve.



Slika 1: Tipi mobilnih naprav.

Pametni telefoni

Med najbolj razširjene mobilne naprave zagotovo sodijo pametni telefoni, ki nam vedno bolj krojijo vsakdanje življenje. Uporabljamo jih za dostop do spleta in pripomočke za vodenje osebnih informacij in družabno mreženje. Ponudba na trgu je silno pestra, število različnih platform pa rekordno za to vrsto izdelkov. Brezžične internetne povezave in zasloni, občutljivi na dotik, so v zadnjih letih povsem spremenili podobo pametnih telefonov. Če smo se še pred slabimi desetimi leti navduševali nad čedalje manjšimi telefoni, ki sodijo v prav vsak žep, je danes, paradoksalno, boljše tisto, kar je večje. A hkrati tanjše, bolj elegantno [7]. Telefoni so postali večpredstavni, internetno povezani terminali in nekakšne večnamenske naprave, ki se iz telefona v hipu spremenijo v računalnik, fotoaparata, video kamero, navigacijsko napravo, večpredstavni predvajalnik (slika 2).



Slika 2: Uporaba pametnih telefonov.

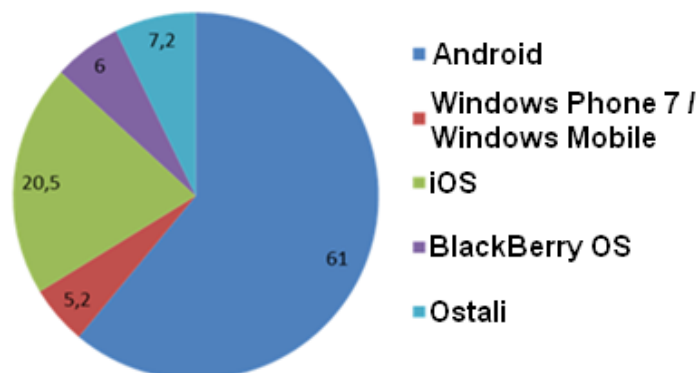
Sodobni pametni telefon je danes po večini lastnosti, razen po velikosti, primerljiv z osebnimi računalniki. Vse več modelov ima procesorje z več jedri, velike pomnilnike in cel niz vmesnikov. Praktično vsi že imajo vmesnike za WiFi, bluetooth, celo GPS. Vsi praktično tudi pokrivajo najsodobnejše mobilne internetne povezave, kot je HSPA. Prihajajo že telefoni naslednje generacije s tehnologijo LTE. Fotoaparati z ločljivostjo pet milijonov pik so vse pogostejši, prav tako imajo možnost snemanja dobrih video posnetkov, celo v visoki ločljivosti. Pojavljajo se prvi modeli, ki imajo dvojne kamere za 3D posnetke [8].

2.2 Mobilne platforme

Mobilne platforme predstavljajo kombinacijo vmesne programske opreme in razvojnih orodij, ki razvijalcem omogočajo preprostejši razvoj na mobilnih napravah. Mobilne platforme se še nadalje intenzivno razvijajo, kar je posledica stalnih zahtev po večji učinkovitosti, zmanjšanju stroškov in večanju produktivnosti zaposlenih. Mobilne platforme omogočajo fleksibilno, varno in učinkovito upravljanje različnih storitev na mobilnih napravah, kot so brskanje po internetu, pošiljanje elektronske pošte, širokopasovni internet, navigacija ipd.

Posamezni proizvajalci mobilnih telefonov ponujajo tudi svojo platformo. Med najbolj razširjene sodijo (slika 3):

- Apple – iOS
- Microsoft – Windows Mobile
- Research In Motion – BlackBerry OS
- Symbian Ltd. – Symbian OS
- Google – Android



Slika 3: Tržni delež mobilnih operacijskih sistemov za leto 2012 [9].

2.2.1 Apple - iOS

iPhone OS (tudi iOS) je platforma s podporo za internet in multimedijo, ki je last podjetja Apple. Operacijski sistem iPhone OS je prvič prišel na trg v napravi iPhone vendar ga danes uporabljajo tudi iPad, iPod Touch in Apple TV. S to potezo se je tudi sistem preimenoval v iOS. Značilen je njegov grafični vmesnik, ki je enostaven in hiter za uporabo ter lepo oblikovan, kar pa za ostale mobilne grafične vmesnike ne velja.



Slika 4: iOS logotip.

Za razvoj aplikacij je podjetje Apple ubralo svojo pot – potrebno je imeti Applov računalnik z MacOS sistemom, saj le na tem teče razvojno okolje Xcode. Razvoj aplikacij je mogoč v jeziku Objective-C, ki je pravzaprav običajni jezik C z dodanimi lastnosti objektnega programiranja povzetega po jeziku SmallTalk.

Sistem je popolnoma zaprt in je namenjen izključno napravam, ki jih izdeluje podjetje Apple, tako imajo razvijalci aplikacij na voljo le omejen dostop do samega sistema, kar podjetje Apple opravičuje s trditvijo, da je to boljše za uporabnika.

Prva generacija mobilnih telefonov iPhone je na ameriški trg prišla poleti 2007, od takrat naprej se širi na vseh svetovnih trgih. Podpira tehnologijo GSM in EDGE ter možnost povezovanja na brezžično internetno omrežje. Za tem so prišle generacije imenovane iPhone 3G, ki ima hitrejši 3G prenos podatkov, tehnologijo UMT ter vgrajen GPS sprejemnik.

Apple že od začetka uporablja veliko odprtokodnih rešitev pri svoji programski opremi, saj tudi iOS temelji na OpenBSD platformi, ki pa je seveda močno obogatena predvsem z uporabniškim vmesnikom, ki je delo podjetja Apple. Brskalniki temeljijo na platformi WebKit in posledično delujejo podobno kot Androidov brskalnik, kljub temu, da so v Applu malenkosti prilagodili način prikazovanja spletne strani [10].

2.2.2 Microsoft - Windows Mobile

Microsoftova platforma Windows Mobile se lahko uporablja na pametnih telefonih in mobilnih napravah. Operacijski sistem temelji na Windows CE 5.2 jedru, ki je že dokaj stara tehnologija, ter predstavlja serijo osnovnih aplikacij, razvitih z uporabo Microsoft Windows API. Tega razvijalec ne vidi, saj ima dostop samo do Silverlight knjižnic.



Slika 5: Windows Mobile logotip.

Operacijski sistem je nekoliko podoben namiznemu operacijskemu sistemu za računalnike. Razvoj aplikacij poteka v okolju Visual Studio ter v vseh podprtih jezikih za .NET ogrodje. Na samem telefonu ni celotnega .NET ogrodja, saj temelji na tehnologiji Silverlight, ki je pravzaprav nekakšna podmnožica .NET ogrodja.

Microsoft je znan, da nikoli ne uporablja tehnologij razvitih izven Microsofta – tako tudi brskalnik ne uporablja WebKit tehnologije kot večina ostalih platform, temveč ima svojo lastno rešitev poimenovano Internet Explorer. To predstavlja za razvijalce velik izziv, saj se brskalnik obnaša v določenih primerih popolnoma drugače kot na drugih platformah in tudi podpora za HTML5 še ni v celoti izvedena. Tako je najšibkejši člen te platforme ravno brskalnik [11].

Tabela 1 prikazuje zgodovino Windows Mobile operacijskih sistemov:

Tabela 1: Operacijski sistem Windows Mobile.

Operacijski sistem	Leto izdaje
Windows CE	1992
Pocket PC 2000	April 2000
Pocket PC 2002	Oktober 2001
Windows Mobile 2003	Junij 2003
Windows Mobile 2003 SE	Marec 2004
Windows Mobile 5	Maj 2005
Windows Mobile 6	Februar 2007
Windows Mobile 6.1	April 2008
Windows Mobile 6.5	Oktober 2009
Windows Phone	Februar 2010

2.2.3 Research In Motion - BlackBerry

V letu 1999 je podjetje RIM (Research In Motion) začelo prodajati mobilne naprave, ki so poleg klasičnih storitev omogočale tudi dostop do e-pošte s pomočjo MIDP 1.0 (Mobile Information Device Profile). Trenutno obstaja različica 2.0, ki omogoča popolno brezžično aktiviranje in sinhronizacijo z Microsoft Exchange, Lotus Domino ali Novell Group Wise. Iz tega se je razvil BlackBerry OS, ki je popolnoma zaprt sistem in omogoča večprocesno komunikacijo, saj ga uporablja le podjetje RIM za svoje naprave.



Slika 6: BlackBerry logotip.

Razvijalci lahko razvijajo programsko opremo s pomočjo API razredov, vendar pa morajo biti mobilne aplikacije, ki vključujejo določene funkcionalnosti, digitalno podprte. BlackBerry je mnogo bolj uporabljen v poslovnih okoljih, saj omogoča celostno rešitev za komuniciranje med zaposlenimi in z drugimi podjetji [12].

2.2.4 Symbian Ltd. – Symbian OS

Začetki segajo v 90. leta prejšnjega stoletja in je eden najstarejših in najbolj uspešnih mobilnih operacijskih sistemov. Podjetje Nokia je vključevala Symbian operacijski sistem v svoje najbolj zmogljive mobilne telefone od leta 2001 dalje.



Slika 7: Symbian logotip.

Razvit je bil v jeziku C v podjetju Symbian, ki je večino kode objavilo pod odprtokodno licenco. Prednost platforme pred drugimi je njena arhitektura, ki je bila razvita za prenosne naprave z majhnimi sistemskimi viri, ki so zahtevale ustrezno robustnost. Pomembna je predvsem nizka poraba pomnilnika, kar omogoča malo možnosti za napake v naslavljanju. Prednost je tudi v tem, da se procesor v primeru neaktivnosti samodejno izklaplja in prehaja v stanje spanja (sleep), kar posledično zmanjša porabo energije [13].

2.2.5 Google - Android

Android je odprtokodna mobilna platforma za pametne telefone ter ostale prenosne naprave, ki vključuje operacijski sistem, vmesno programsko opremo in ključne mobilne aplikacije. Glavna razlika med Googlovim Android sistemom v primerjavi z ostalimi je v tem, da je omogočen razvoj aplikacij drugim razvijalcem in so narejene enakovredno privzetim aplikacijam. Do nedavnega so imeli mobilni telefoni nameščene lastniške (»zaprti kodni«) licenčne operacijske sisteme, ki so zahtevali licenčno programsko opremo za razvoj mobilnih aplikacij. Proizvajalci telefonov so dajali prioriteto privzetim aplikacijam pred tistimi, napisanimi s strani drugih razvijalcev, kar je negativno vplivalo na razvoj aplikacij. Na Androidu so vse aplikacije razvite z enakimi knjižnicami in se izvajajo na enotnem izvajalnem okolju, kar omogoča razvijalcem razviti kompleksne aplikacije, ki zahtevajo zmogljivejšo strojno opremo, posledično prihaja do hitrejšega razvoja mobilnih naprav [14].







Slika 8: Android logotip.

Podjetje Android Inc. So ustanovili Andy Rubin, Rich Miner, Nick Sears in Cris Whiteeta, leta 2003 v Kaliforniji, v mestu Palo Alto z namenom razvoja naprednejših mobilnih naprav, ki se bodo lahko prilagodile posameznikovim potrebam in željam. Leta 2005 je lastništvo nad podjetjem prevzelo podjetje Google, vključno z zaposlenimi in ustanovitelji. S to potezo je Google želel vstopiti na mobilni trg, zato so takrat začeli z razvojem operacijskega sistema za mobilne naprave, ki bi temeljil na Linux jedru.

Novembra 2007 so objavili prihod odprtokodne mobilne platforme Android z ustanovitvijo združenja imenovano Open Handset Alliance (OHA) [15]. V to združenje so danes vključena velika podjetja (Motorola, T-Mobile, Samsung, Sony Ericsson, Toshiba, Vodafone, Google, Intel, Texas Instruments) s področja proizvodnje računalniških čipov, programske opreme, mobilni operaterji in ponudniki mobilnih telefonov. Oktobra 2008 je Google ponudil izvorno kodo pod okriljem odprto kodne licence Apache. Na koncu leta 2008 je sledil prvi popravek za Android OS, aprila leta 2009 pa je bil izdan še SDK 1.5, v katerem je bilo dodanih veliko novih funkcionalnosti.

Imena različic so poimenovana po sladica, ki si sledijo po abecednem redu. Njihove pomembnejše izboljšave so prikazane v tabeli 2 [16].

Tabela 2: Android različice.

Verzija	Dodatki/popravki
Android 1.0	23. september 2008, začetna verzija. Budilka, kamera, računalnik, slike, SMS, MMS, spletni brskalnik, e-pošta (podpora POP3, IMAP4 in SMTP), Gmail sinhronizacija, Android Market, Google Maps, Google Talk, Google Contacts, Google Sync, Google Calendar, Google Search, Media Player, YouTube, osnovne aplikacije, WiFi, Bluetooth.
Android 1.1	9. februar 2009 Zaslonska tipkovnica, posodobljen Google Maps, shranjevanje priponk pri sporočilih, ugašanje zaslona med telefoniranjem.
Android 1.5  CUPCAKE	30. april 2009 (Linux 2.6.27) Videokamera, predvajanje MPEG-4 in 3GP, prikazovanje slik uporabnika, posebna ura in datum, nalaganje videoposnetkov na YouTube, nalaganje fotografij Picasa, animirani zaslonski prehod, večjezična podpora.
Android 1.6  DONUT	15. september 2009 (Linux 2.6.29) Pretvorba besedila v govor, posodobljen Android Market, podpora za WVGA zaslone, izboljšana podpora za razvijalce (SDK, AVD). Na HTC Hero so skupaj z Open Handset Alliance zagnali to različico operacijskega sistema.
Android 2.0  ECLAIR	26. oktober 2009 (Linux 2.6.29) Posodobitev e-pošte (več računov hkrati), dodelana SMS in MMS (brisanje starih ko je pomnilnik poln), fotoaparati (zoom, bliskavica), pohitritev operacijskega sistema, nov uporabniški vmesnik, Bluetooth 2.1, brskalnik (HTML5), Google Maps 3.1.2, večtočkovno zaznavanje.
Android 2.0.1  ECLAIR	3. december 2009 Odprava napak pri verziji 2.0

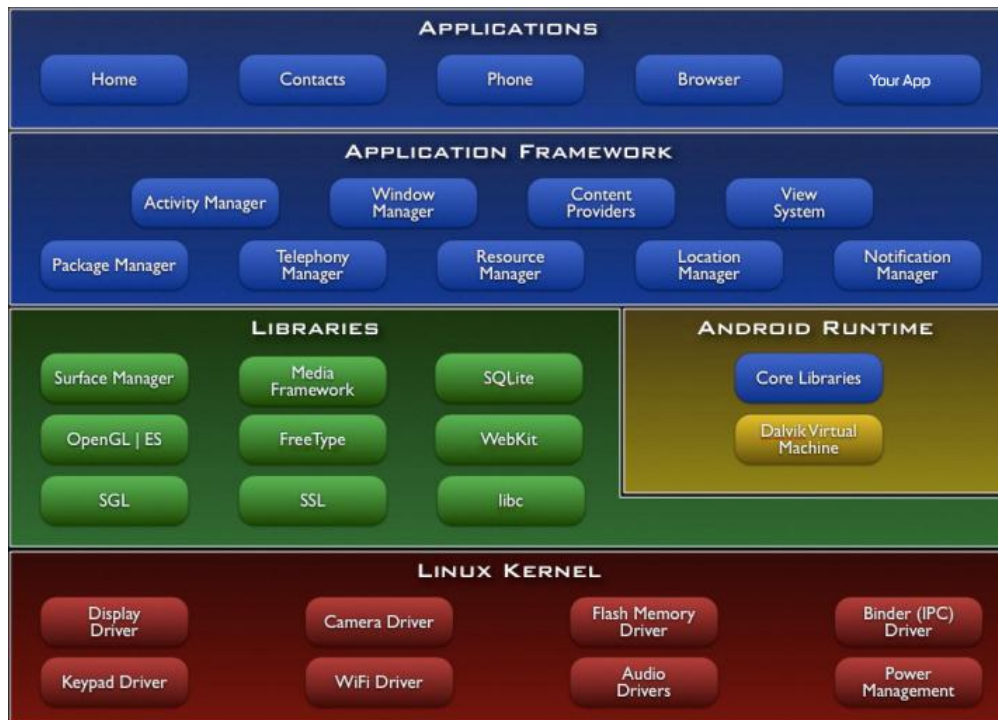
<p>Android 2.1</p>  <p>ECLAIR</p>	<p>12. januar 2010</p>
<p>Android 2.2</p>  <p>FROYO</p>	<p>20. maj 2010 (Linux 2.6.32) Nameščanje programov na spominsko kartico, integriranje Chrome z JavaScript, vizualno dodelan uporabniški vmesnik, podpora Microsoft Exchange, USB povezovanje, Adobe Flash, sistemsko odlagališče, WiFi hotspot, Android Market omogoča samodejno posodobitev nameščenih aplikacij.</p>
<p>Android 2.3</p>  <p>GINGERBREAD</p>	<p>6. december 2010 Novi senzorji (giroskop, barometer), posodobljen uporabniški vmesnik (hitrejši in enostavnejši), orodja za kopiranje in lepljenje elementov, podpora za veliko ločljivost (WXGA), SIP VoIP, izboljšana virtualna tipkovnica, Bass Boost, upravljanje porabe energije, video format VP8, Download Manager.</p>
<p>Android 3.0</p>  <p>HONEYCOMB</p>	<p>22. februar 2011 (Linux 2.6.36) Hiter dostop do obvestil, optimiziran uporabniški vmesnik, prilagojen napravam z večjimi zasloni, izboljšana 2D in 3D grafika, galerija v celozaslonskem načinu, video klepet Google Talk, izboljšana strojna oprema, podpora za večjedrne procesorje.</p>
<p>Android 3.1</p>  <p>HONEYCOMB</p>	<p>10. maj 2011 Izboljšan uporabniški vmesnik, povezovanje preko USB, podpora za zunanjo tipkovnico, visokozmogljiv WiFi.</p>
<p>Android 4.0</p>  <p>ICE CREAM</p>	<p>19. oktober 2011 Strojno pospešen in prenovljen grafični vmesnik, odklepanje s prepoznavanjem obraza, nov spletni brskalnik, izboljšana aplikacija za kamero, dodan nov glasovni vnos.</p>
<p>Android 4.1</p>  <p>JELLY BEAN</p>	<p>Varnostni in programski popravki.</p>

Arhitektura

Android operacijski sistem je sestavljen iz petih glavnih elementov (slika 9):

- aplikacije (ang. Applications),
- aplikacijsko ogrodje (ang. Application framework), vsebuje vse sistemske aplikacije,
- izvajalno okolje (ang. Android Runtime),

- knjižnice (ang. Libraries), temeljni del operacijskega sistema,
- jedro Linux (ang. Linux Kernel), vsebuje gonilnike za podporo strojni opremi.



Slika 9: Android arhitektura [17].

Aplikacije

Android operacijski sistem ločuje dve vrsti aplikacij, primarne aplikacije, kot so koledar, imenik stikov, odjemalec za elektronsko pošto, brskalnik, SMS aplikacija, galerija slik, aplikacija za video kamero, predvajalnik glasbe, kalkulator, namizje, budilka, ipd. Te aplikacije so nameščene že od samega začetka. Aplikacije razvite s strani drugih razvijalcev, ki jih lahko poljubno namestimo z uporabo Android Marketa pa imenujemo sekundarne aplikacije. V veliko primerih pa imamo nameščene Googlove aplikacije.

Aplikacije so razvite v programskem jeziku Java, kjer Android SDK prevede izvorno kodo in vire v skupno datoteko *.apk. Osnovni gradniki aplikacij so aplikacijske komponente, kjer vsaka komponenta igra drugačno vlogo pri obnašanju aplikacije in je lahko aktivirana posebej, tudi s strani drugih aplikacij. Poznamo štiri različne tipe komponent: aktivnost, storitev, ponudnik vsebin, sprejemnik za razpršeno oddajanje [18].

Aplikacijsko ogrodje

Aplikacijsko ogrodje omogoča razvijalcem aplikacij, da lahko dostopajo do strojne opreme in s tem do raznih informacij, nastavite-v, ipd. Razvijalci imajo popoln dostop do enakega API ogrodja, kot ga uporabljajo primarne aplikacije. Ogrodje nam omogoča, da bodo aplikacije prikazane enako na vseh mobilnih napravah s podporo operacijskega sistema Android [18].

- Upravitelj aktivnosti (ang. Activity Manager), nadzira življenjski cikel aktivnosti.
- Upravitelj obveščanj (ang. Notification Manager), opozarja in obvešča uporabnika v statusni vrstici.
- Ponudnik vsebine (ang. Content Provider), omogoča aplikacijam izmenjavo podatkov.
- Pogledi (ang View), uporabniški vmesnik, ki vsebuje gube, tekstovna polja, mreže in tudi brskalnik.

Knjižnice

Poleg aplikacijskega ogrodja vsebuje Android še zbirko programskih knjižnic, namenjenih uporabi različnih delov samega sistema [18]. V naboru knjižnic so:

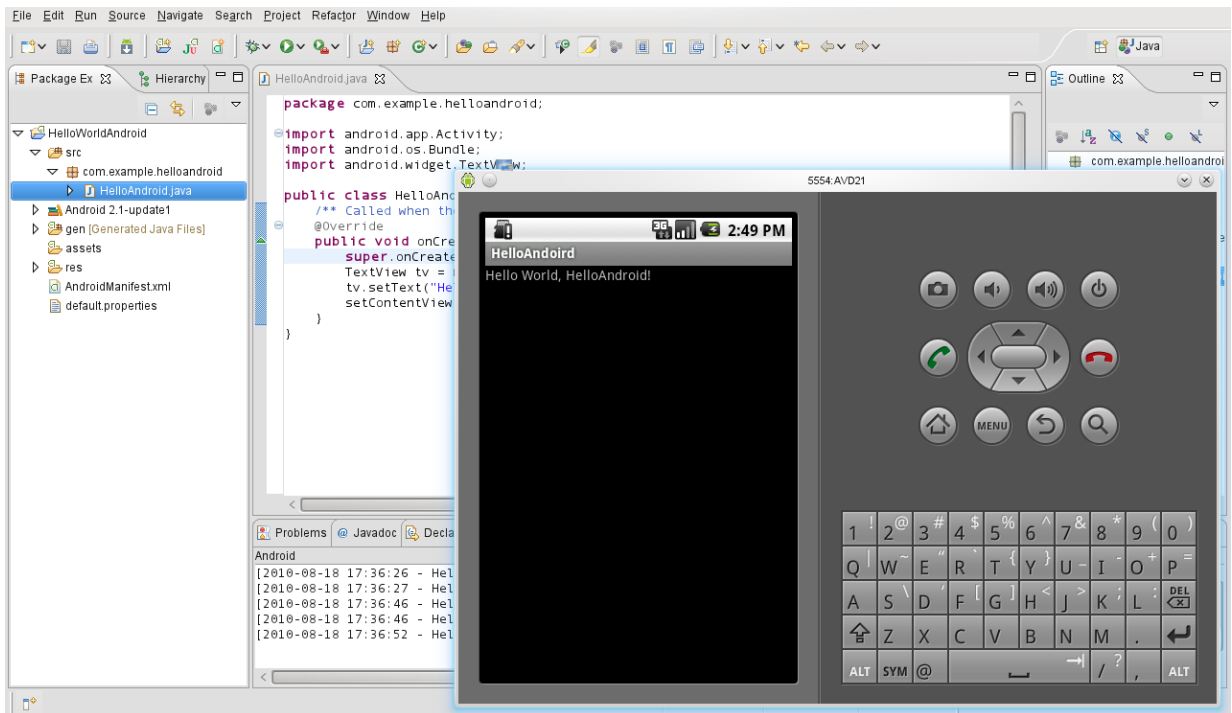
- Večpredstavna knjižnica temelji na knjižnicah paketa PacketVideo OpenCORE, namenjene predvajanju in snemanju številnih priljubljenih večpredstavnih vsebin, vključno z MPEG4, H.264, MP3, ACC, AMR, JPG in PNG
- Sistemska knjižnica C, ki predstavlja za mobilne naprave prilagojeno standardno sistemsko knjižnico libc.
- Upravitelj površin (Surface Manager), ki upravlja dostop do zaslonskega podsistema in sestavlja 2D in 3D sloje aplikacij.
- LibWebCore in WebKit, predstavljata vgrajen brskalnik in podporne komponente. Knjižnico uporabljata tudi brskalnika Chrome in Safari.
- SGL, povezava z osnovnim slojem 2D grafičnega pogona, 3D knjižnice pa povezavo s podsistemom OpenGL ES, ki skrbi za izris 3D grafike.
- Knjižnice FreeType, namenjena prikazu bitne in vektorske pisave.
- SQLite, predstavlja v sistem vključeno relacijsko zbirko podatkov, katero uporablja tudi Firefox in iPhone.

Linux jedro

Samo osrčje Androida predstavlja jedro, ki zagotavlja varnost, upravljanje s pomnilnikom, upravljanje s procesi, omrežni sklad in model gonilnikov. Linux jedro predstavlja neke vrste abstrakcijo med strojno opremo in programskim skladom, njegovi odliki pa sta zanesljivost in robustnost.

2.3 Eclipse

Razvojno okolje Eclipse odlikuje predvsem sistematičnost grafičnega vmesnika, ki je silno pregleden in funkcionalen. Možnosti je v osnovni različici dovolj za osnovno delo, za kaj bolj specifičnega pa je treba poseči po dodatkih (ang. plugins) - teh je v svetovnem spletu skorajda neomejeno. Tako je na primer mogoče napolniti okolje s številnimi zgledi, dodati podporo XML ali pa vgraditi Visual Editor za grafične programe (torej javanske grafične vmesnike).



Slika 10: Razvojno okolje Eclipse.

Bistvo uporabniškega vmesnika okolja Eclipse je uporaba tako imenovanih perspektiv. Vsaka perspektiva je svoja množica pogledov (ang. views) in urejevalnikov ter predstavlja prilagoditev samega okolja določenemu namenu oziroma vrsti dela. Tako na primer z odprtjem perspektive Debug spremenimo skorajda celoten videz okolja - spremenijo se pogledi, konzola, dobimo nove gumbe in možnosti za iskanje napak oziroma hroščev. Skratka, vsaka perspektiva nosi s seboj vse potrebno za določeno vrsto opravil. Osnovna oziroma privzeta perspektiva je javanska perspektiva, ki je osnova za pisanje same kode (v njej imamo torej urejevalnik javanske kode, konzolo, navigator po projektih, paketih in razredih, preglednik objektov, metod in spremenljivk...). Poleg tega imamo tudi perspektivo za ekipno delo (CVS repository), perspektivo za preiskovanje javanskih projektov, perspektivo za razvoj dodatkov Eclipse itd. Tako načelo perspektiv imajo na voljo tudi plačljiva orodja, ki temeljijo na Eclipsu, le da je tam perspektiv še veliko več (pri orodjih za J2EE razvoj celo takšne za delo z aplikacijskim strežnikom, zbirkami podatkov itd.) [19].

2.4 Programska orodja

Programska orodja se uporabljajo za razvoj aplikacij. Za podatke se uporabljajo zapisi v obliki XML (ang. Extensible Markup Language) za programiranje Android aplikacij, pa potrebujemo Android SDK.

XML

XML je razširljiv označevalni jezik, ki je podoben HTML-ju in nam omogoča format za opisovanje in tudi shranjevanje strukturiranih podatkov ali arhitektura za prenos podatkov in njihovo izmenjavo med več omrežji. XML spreminja mnogo aspektov računalništva, še posebej na področju komuniciranja aplikacij in strežnikov. Da pa se ga tudi razširiti, saj ima namreč to možnost, da si lahko sami izmislimo imena značk (ang. tag). Zelo je uporaben, saj ima zelo preprosto in pregledno zgradbo [20].

Razdeljen je na 3 dele:

- podatkovni (vanj shranimo podatke v neki obliki z željenimi značkami (ang. tag)),
- deklarativni (skrbi za to, da lahko pri dodajanju novih podatkov vidimo kaj kakšna etiketa predstavlja),
- predstavitevni (z njim oblikujemo izpis podatkov).

Android SDK

Android SDK je skupek razvijalskih orodij, ki so potrebna za razvoj aplikacije v programskem jeziku Java. Sestavljen je iz razhroščevalnika, knjižnic, emulatorja, dokumentacije, primerkov kode in vodičev. Razvojno okolje podpira računalnik, ki ga poganja Linux, Mac OS X 10.4.9 ali novejši in Windows XP ali novejši operacijski sistem. Razvojno okolje (IDE) je Eclipse in uporablja vtičnik ADT, ki pa ni edina možnost za razvoj aplikacij. Uporabimo lahko katerikoli urejevalnik besedil za Javo in XML.

Android SDK je povsem brezplačen in se ga namesti s spletne strani, namenjene razvijalcem androida. Za zagotavljanje kompatibilnosti Android SDK podpira tudi starejše verzije androidovih platform, saj je razvoj SDK orodij sorazmeren z razvojem platforme [21].

3 Razvoj mobilne aplikacije

V tem poglavju podrobneje predstavimo razvoj mobilne aplikacije za platformo Android. Aplikacija je namenjena za prikaz senzorskih meritev, ki so shranjene na spletnem strežniku. Opišemo vse tri tipe datotek. Naredimo načrt aplikacije, katerega prikažemo s shemo, podrobneje predstavimo vsa zaslonska okna, ki se nam lahko pojavijo med delovanjem in na koncu opišemo potek in izdelavo logotipa in grafik.

3.1 Opis senzorskih meritev

V aplikaciji dostopamo do treh različnih datotek, ki so v obliki XML. Ob zagonu aplikacije preberemo datoteko, kjer so shranjene povezave do datotek s podatki o temperaturah. Datoteka vsebuje korenski element z imenom *files* in podelemente z imenom *file*, iz katerih dobimo povezave do datotek z vrednostmi o temperaturah (slika 11). V primeru da se na spletni strežnik naloži nova datoteka z temperaturami, se v tej datoteki naredi nov podelement *file*. Dopolnitev te datoteke vzdržuje spletna storitev, ki je neodvisna od moje aplikacije.

```
<files>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120518-141034-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120608-160511-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120518-140837-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/111022-080000-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120605-160405-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120605-160330-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120423-094330-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120525-125146-AG11-sensor.xml</file>
  <file>http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files/120325-073000-AG11-sensor.xml</file>
</files>
```

Slika 11: Podatki o datotekah.

V datoteki kjer so zapisane vrednosti o temperaturah imamo korenski element *Sensor*, ki vsebuje atribut *uri* in nam poda številko podatkovnega beležnika s katerim so bile narejene meritve (slika 12). Podelement *Temperature* vsebuje atribut *at* iz katerega dobimo dan in uro meritve. Oblika zapisa je *leto-mesec-dan+ura:minuta:sekunda*. Vrednost podelementa *Temperatura* je izmerjena temperatura.

```
<Sensor uri="383004776/asset/004/7">
  <Temperature at="2012-04-20+08:13:29">26.8</Temperature>
  <Temperature at="2012-04-20+08:19:29">26.8</Temperature>
  <Temperature at="2012-04-20+08:25:29">26.98</Temperature>
  <Temperature at="2012-04-20+08:31:29">26.98</Temperature>
  <Temperature at="2012-04-20+08:37:29">26.8</Temperature>
  <Temperature at="2012-04-20+08:43:29">26.62</Temperature>
  <Temperature at="2012-04-20+08:49:29">26.62</Temperature>
  <Temperature at="2012-04-20+08:55:29">26.62</Temperature>
  <Temperature at="2012-04-20+09:01:29">26.44</Temperature>
  <Temperature at="2012-04-20+09:07:29">26.26</Temperature>
  <Temperature at="2012-04-20+09:13:29">26.44</Temperature>
  <Temperature at="2012-04-20+09:19:29">26.44</Temperature>
  <Temperature at="2012-04-20+09:25:29">26.44</Temperature>
</Sensor>
```

Slika 12: Podatki o temperaturah.

Do podatkov o sledljivosti dostopamo preko spletne storitve z ukazom »*številka čitalnika*?date=leto-mesec-dan. Številko čitalnika in datum, ki je potreben za zagon spletne storitve preberemo iz datoteke o temperaturah. Korenski element *data* vsebuje podelement *event* s tremi atributi. Atribut *time* v obliki *leto-mesec-dan+ura:minuta:sekunda* nam poda datum in uro začetka akcije, ki je zapisana v atributu *bizstep*. Zadnji atribut *disp* nam poda lokacijo kjer se je akcija izvedla (slika 13).

```
<data>
  <event time="2012-08-23+15:06:08" bizstep="Pakiranje" disp="Sortirni prostor"/>
  <event time="2012-08-23+15:16:08" bizstep="Izdaja iz sortirnice" disp="Sortirni prostor"/>
  <event time="2012-08-23+16:20:35" bizstep="Prevzem v hladilnici" disp="V/I vrata hladilnice"/>
  <event time="2012-08-24+05:33:35" bizstep="Izdaja iz hladilnice" disp="V/I vrata hladilnice"/>
  <event time="2012-08-24+13:39:54" bizstep="Prevzem v ribarnici" disp="Vrata ribarnice"/>
</data>
```

Slika 13: Podatki o sledljivosti.

3.2 Načrt aplikacije

Najprej smo si izdelali načrt razvoja aplikacije, ki se je skozi čas ravoja aplikacije malo spremenil. Osnovna funkcionalnost aplikacije je prikazovanje senzorskih meritev, ki so shranjene na spletnem strežniku. Za realizacijo takšnega scenarija smo določili dva načina prikaza senzorskih podatkov.

V prvem načinu bi podatke prikazali s pomočjo razreda *ListView*, ki ga napolnimo s poljem nizov z uporabo adapterja. Polje nizov smo definirali in napolnili s pridobitvijo podatkov iz XML dokumentov na spletnem strežniku. Druga možna rešitev prikaza podatkov pa bi bila izris črtnega grafa s pomočjo knjižnice *AChartEngine 1.0.0*, ki je zunanja knjižnica za risanje grafov. Omenjena knjižnica ima široko paleto raznovrstnih grafov, kot so palčni, tortni, črtni,... Z nastavljivimi parametri, od območja na oseh, velikosti pisave, barve različnih elementov grafa ipd.

Dostopamo do treh različnih XML datotek, ki so lahko zelo velike, zato smo si pri branju in razčlenjevanju dokumentov pomagal z knjižnico SAX API. SAX API ne prebere celotnega dokumenta XML v spomin, zato je primeren za delo z velikimi - ogromnimi datotekami. Poleg tega omogoča, da hranimo podatke v kompleksni podatkovnih strukturah, ki jih definiramo v aplikaciji. SAX API je enostaven za uporabo in temelji na dogodkih. To pomeni, da razčlenjevalnik bere dokument in sporoči programu, ko najde določen simbol – na primer začetek ali konec elementa določene znakovne podatke, ipd. Spodaj je prikazan primer branja začetnega in končnega elementa iz datoteke, ki vsebuje vrednosti o temperaturah.

```
@Override
public void startElement(String uri, String localName,
    String qName, Attributes attributes) throws SAXException {
    currentElement = true;
    if (localName.equals("Sensor")){
        tempList = new TempList();
        String uriS = attributes.getValue("uri");
        tempList.setUriS(uriS);
    } else if (localName.equals("Temperature")) {
        String attr = attributes.getValue("at");
        tempList.setAt(attr);
    }
}
@Override
public void endElement(String uri, String localName, String qName)
    throws SAXException {
    currentElement = false;
    if (localName.equalsIgnoreCase("Temperature"))
        tempList.setTemp(currentValue);
}
```

Aplikacijo smo razvil tako, da bi jo bilo možno kasneje posodobljati in nadgrajevati, zato smo si najprej narisali osnovno shemo aplikacije, ki vsebuje vse potrebne glavne gradnike aplikacije in tudi predvideno funkcionalno delovanje aplikacije. Aplikacija naj bi tako vsebovala štiri različne zaslone. Za pomikanje med zaslone smo uporabil vizualni gradnik, imenovan *TabHost*, ki vsebuje dve komponenti:

- *TabWidget*: Vsebuje oznake in ikone posameznih zavihkov. S klikom na izbran zavihek se na t.i. starševski kontroli, ki je v tem primeru *TabHost*, proži dogodek. V obravnavi dogodka poteče menjava vsebine, ki je povezana z zaznamkom, ki je dogodek prožil.
- *FrameLayout*: Vsebuje vsebino, ki je povezana s posameznimi zaznamki. Gradniki, ki predstavljajo vsebino, so lahko pogledi (ang. View) ali aktivnosti (ang. Activity). Le-ti se v obliki sklada nalagajo eden preko drugega. Velikost celotne komponente je odvisna od velikosti gradnika, ki zavzame največ prostora.

Spodaj je podana programska koda razreda *TabLayoutSample*, ki nam naredi tri zavihke, katere uporabljamo v aplikaciji.

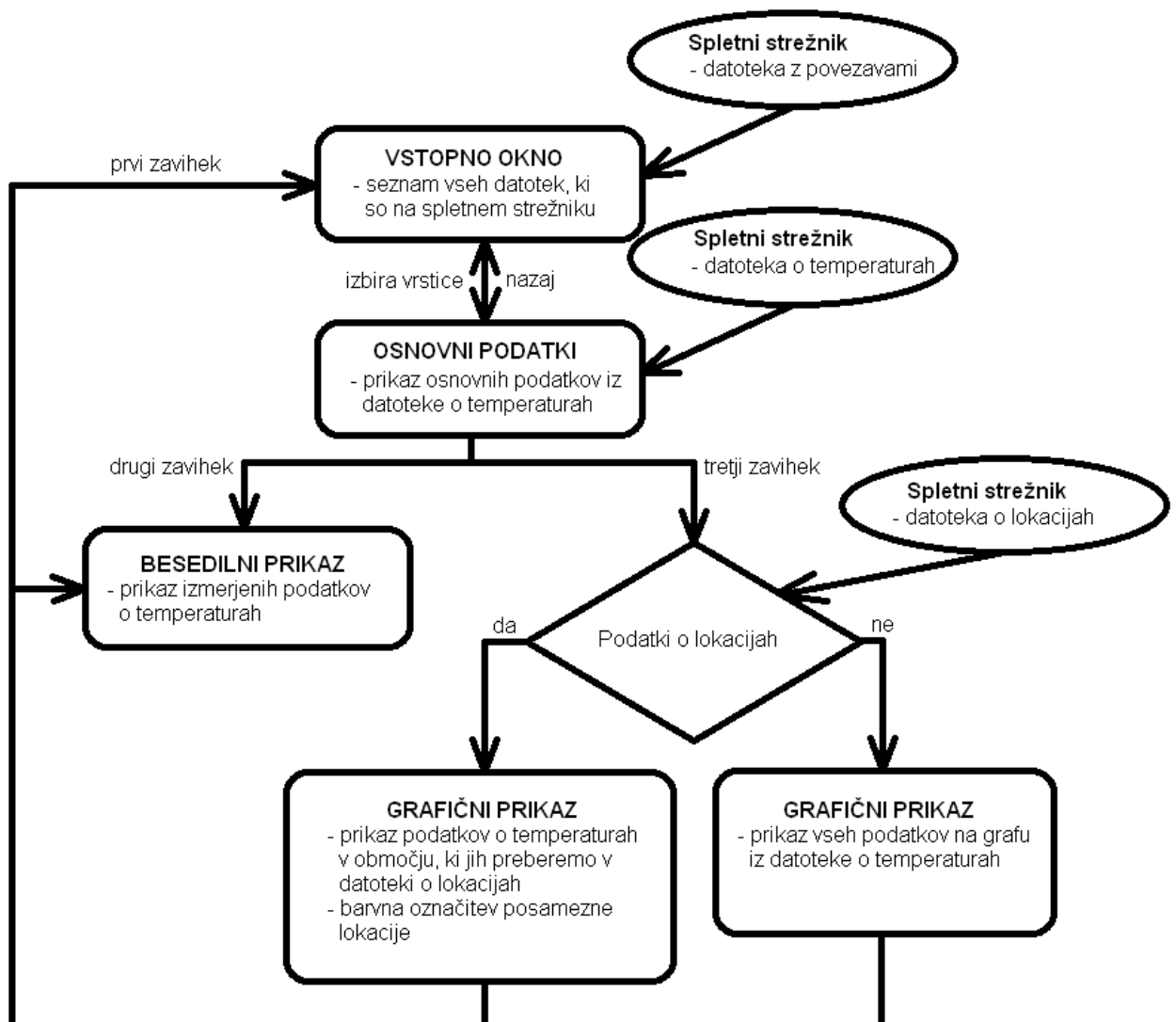
```
public class TabLayoutSample extends TabActivity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        Resources res = getResources();
        TabHost tabHost = getTabHost();
        TabHost.TabSpec spec;
        Intent intent;
        // Vstopno okno
        intent = new Intent().setClass(this, HomeGroup.class)
            .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        spec = tabHost.newTabSpec("nastavitve").setIndicator(
            "",res.getDrawable(R.drawable.tab_nastavitve))
            .setContent(intent);
        tabHost.addTab(spec);
        // Izpis temperatur
        intent = new Intent().setClass(this, List.class)
            .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        spec = tabHost.newTabSpec("besedilo").setIndicator(
            "",res.getDrawable(R.drawable.tab_besedilo))
            .setContent(intent);
        tabHost.addTab(spec);
        // Grafični prikaz
        intent = new Intent().setClass(this, Graph.class)
            .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        spec = tabHost.newTabSpec("graf").setIndicator(
            "",res.getDrawable(R.drawable.tab_graf))
            .setContent(intent);
    }
}
```

```

tabHost.addTab(spec);
tabHost.setCurrentTab(0);
}
}

```

Ob zagonu aplikacije se nam odpre vstopno okno, na katerem se izpišejo vse povezave do datotek o temperaturah. Uporabnik nato izbere vrstico za katero želi pregledati podatke o temperaturah. Izbira lahko med vsemi datotekami, ki so zapisane na spletnem naslovu <http://ira-f2f.fri.uni-lj.si:8080/assets/resource/files> in jo ob zagonu razčlenimo, tako da nam vsak zapis predstavlja novo vrstico na zaslonu. Ob kliku na posamezno vrstico se odpre zaslon, kjer se izpišejo osnovni podatki, ki so zapisani v datoteki s temperaturami. Z izbiro prvega zavihka ali tipko nazaj na mobilnem telefonu se pomaknemo na vstopno okno kjer imamo možnost ponovne izbire datoteke. Drugi zavihek nam omogoča prikaz podatkov v besedilnem načinu, tretji pa izris grafa. Shema funkcionalnega delovanja aplikacije je predstavljena na sliki 14.



Slika 14: Shema funkcionalnega delovanja aplikacije.

3.3 Pridobitev in prikaz podatkov

Vstopno okno

Vstopno okno je okno, ki se pojavi ob zagonu aplikacije, izpišejo se nam podatki, do katerih dostopamo s povezavo <http://lra-f2f.fri.uni-lj.si:8080/assets/resource/files>. Datoteko razčlenimo tako, da nam vsak element v datoteki predstavlja novo vrstico v aplikaciji, ki se začne z *Temp_*. Prikaz podatkov je strukturiran tako, da so najnovejši zapisi do datotek postavljeni na vrhu, do ostalih zapisov pa pridemo s premikom prsta po ekranu (slika 15).



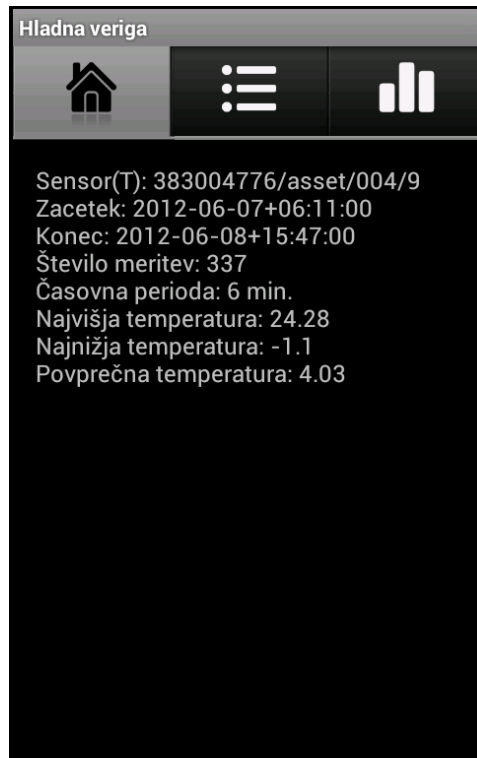
Slika 15: Vstopno okno.

Povzetek datoteke

Okno se nam prikaže, ko izberemo eno od datotek, ki so prikazane v vstopnem oknu. Prikažemo osnovne podatke, ki jih vsebuje datoteka o temperaturah, to so (slika 16):

- Številko oz. oznako čitalnika s katerim so bili podatki prebrani,
- začetek in konec meritve,
- število izvedenih meritev,
- časovna perioda in
- najvišja in najnižja in srednja vrednost temperatur.

Tako dobimo hiter pregled podatkov, ki jih lahko z izbiro na zavihkih prikažemo v besedilnem ali grafičnem načinu. V primeru, da želimo izbrati drugo datoteko se s tipko nazaj na telefonu pomaknemo na vstopno okno.



Slika 16: Povzetek datoteke.

Izpis temperatur

Do izpisa temperatur pridemo z izbiro drugega zavihka, kjer se nam prikažejo podatki, iz izbrane datoteke na vstopnem oknu. Prikaz podatkov je strukturiran v obliki, ki se začne s številko zapisa, datumom, uro meritve in izmerjeno temperaturo. Vrednosti temperature, ki so večje od štiri stopinje in manjše od nič stopinj so obarvane rdeče, ostali zapisi so obarvani zeleno (slika 17). Tako dobimo hiter pregled za vrednosti, ki so izven meja. S premikom prsta se pomikamo po ekranu do prikaza vseh vrednosti.

V primeru, da izberemo izpis temperatur pred izbiro datoteke o temperaturah na vstopnem oknu se nam pojavi sporočilno okno razreda *Toast*. Sporočilno okno vsebuje besedilo »Nobena datoteka ni bila izbrana.« in nastavljen kratek čas prikaza. Pozicija postavitve prikaza je nastavljena na sredino zaslona.

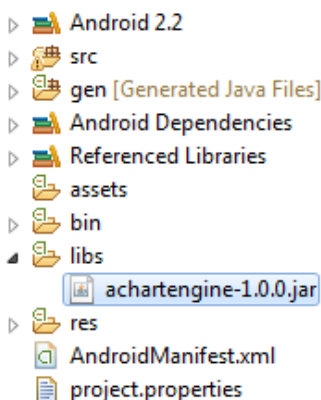
```
Toast toast = Toast.makeText(this, "Nobena datoteka ni bila izbrana.",
    Toast.LENGTH_SHORT);
toast.setGravity(Gravity.CENTER | Gravity.CENTER, 0, 0);
toast.show();
```



Slika 17: Izpis temperatur.

Grafični prikaz

Za pomoč pri izrisu grafa smo si pomagali z brezplačno knjižnico AChartEngine 1.0.0 pod licenco Apache Licence, verzija 2.0, ki je namenjena za risanje grafa in jo lahko uporabljamo tudi v aplikacijah, ki jih želimo uporabljati v komercialne namene. Knjižnica Podpira 1.6 različice Android SDK in novejše. Vse zunanje knjižnice je potrebno shraniti v mapi libs, ki jo ustvarimo v korenu projekta (slika 18).



Slika 18: Datotečna struktura.

Grafični izris izvaja gradnik *GraphicalView*, ki črpa podatke iz razreda *XYSeries*, tabelarične predstavitev podatkov. Poleg njiju so še tu pomožni nadzorni razredi za raznolike nastavitve, kot je *XYSeriesRenderer* za slog izrisa.

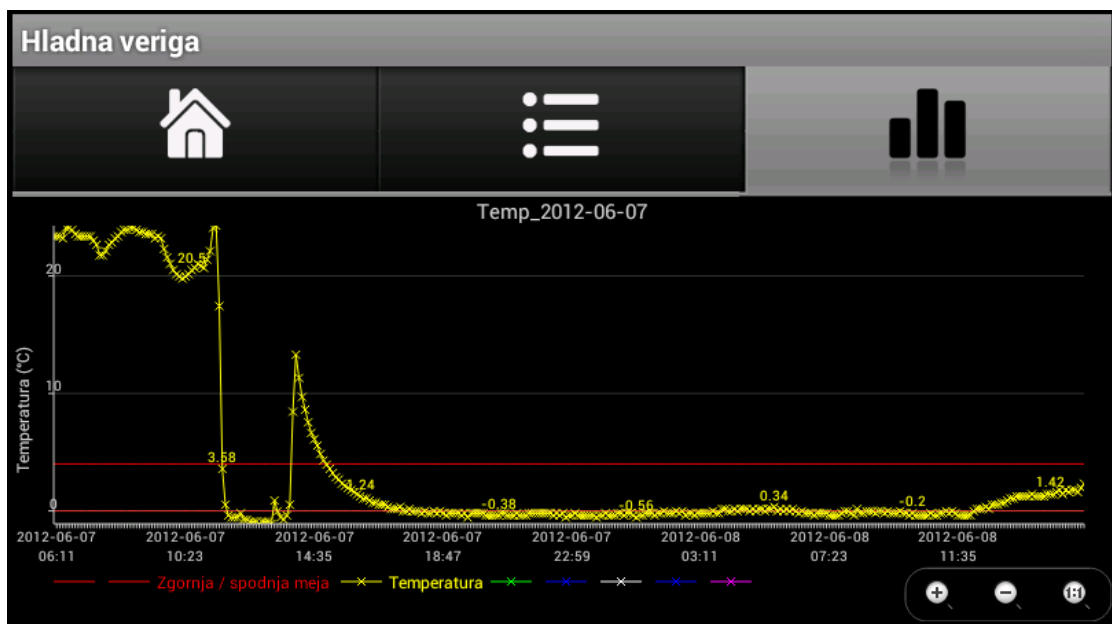
Zadnji zavihek nam prikaže podatke v grafičnem načinu, kjer se nam izriše graf. Na y osi prikazujemo podatke o temperaturah kjer zgornja vrednost postane največja prebrana vrednost, spodnja vrednost pa najmanjša vrednost. Na x osi so prikazani podatki o datumu in uri meritve. Na vrhu je prikazan tekst izbrane datoteke iz vstopnega okna. Na grafu vedno prikažemo dve horizontalni rdeči črti, ki predstavljata kritično mejo temperature. Zgornja črta prikazuje štiri stopinje, spodnja pa nič stopinj. Ob kliku na posamezno točko na grafu, se nam pojavi sporočilno okno, ki nam prikaže dan, uro meritve in izmerjeno vrednost. Tako lahko dobimo točno informacijo za posamezno točko na grafu. Graf lahko tudi s pomočjo gumbov ki so v spodnjem desnem kotu tudi približujemo in oddaljujemo in natančneje pregledamo vrednosti za posamezno območje.

Pri izpisu grafa dostopamo do spletne storitve, ki nam z atributoma, številka podatkovnega beležnika in imenom datoteke vrne podatke o sledljivosti, če jih ima, tako dobimo dva načina izpisa grafa:

- a. brez sledljivosti in
- b. s sledljivostjo.

Brez sledljivosti.

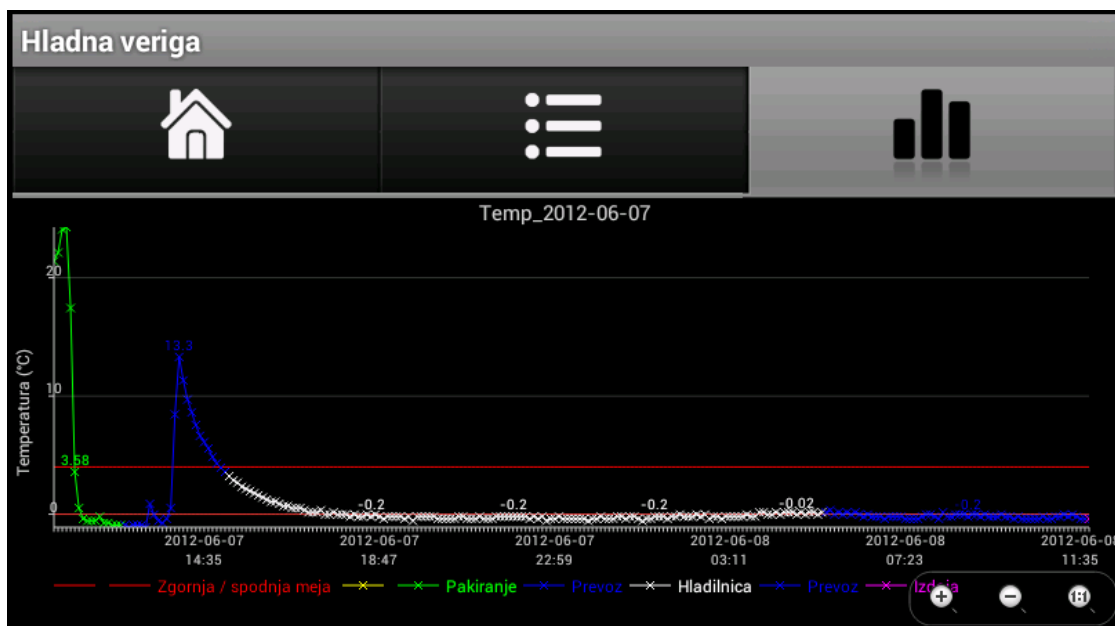
V tem primeru nam spletna storitev ne vrne podatka o sledljivosti in graf izrišemo z vsemi podati, ki jih vsebuje datoteka s temperaturami. Graf je v celoti obarvan rumeno (slika 19).



Slika 19: Izris grafa brez podatkov o sledljivosti.

S sledljivostjo

V primeru, da imamo za izbran graf na voljo tudi podatke o sledljivosti, se nam posamezni deli grafa obarvajo z drugačno barvo, ki prikazuje določeno akcijo. Zeleno obarvana krivulja ponazarja obdobje ko je bilo izvedeno pakiranje, modra označuje prevoz, bela hladilnica, roza pa čas izdaje (slika 20). Tudi v grafičnem načinu se nampojavi sporočilno okno z isto vsebino in obliko kot pri izpisu temperatur v primeru, da želimo izrisati graf preden izberemo datoteko s temperaturami.



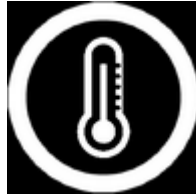
Slika 20: Izris grafa z podatki o sledljivosti.

3.4 Izdelava logotipa in grafik za gumbe

Logotip za aplikacijo smo izdelal v programu Adobe Photoshop verzije CS5. Izdelal smo tudi grafike za gumbe, ki so v zavihkih. Najprej smo na list papirja narisal skico logotipa, ki smo ga nato ustvaril v programu Adobe Photoshop. Logotip je sestavljen iz dveh delov: belega kroga in znaka temperatura na črni podlagi (slika 21).

Držali smo se treh priporočljivih elementov dobrih ikon:

- Enostavnost: ikone naj bodo poenostavljene, celo karikirane. Primerne za uporabo v majhnih velikostih.
- Modernost: ikone naj imajo sodobno obliko.
- Pogled od spredaj: Ikone naj bodo obrnjene naprej, uporaba perspektive ni potrebna.



Slika 21: Logotip aplikacije.

Izdelal semo še tri grafike za zavijke, ki smo jih potrebovali pri razvoju grafičnega vmesnika. Naredil smo tri elemente: hišico, vodoravne črte in tri stolpce. Vse tri grafike se ob izbiri pobarvajo črno na sivi podlagi z osvetlitvijo od zgoraj navzdol. V primeru da grafika ni izbrana so elementi obarvani z belo barvo na črni podlagi (slika 22).



Slika 22: Grafika za gumbe.

4 Test aplikacije

Aplikacijo smo testirali v dveh delih. Najprej smo aplikacijo testirali na Emulatorju Android 2.2 v razvojnem okolju Eclipse, nato pa še na mobilni napravi. Aplikacijo smo testirali z večimi testnimi primeri, da smo pridobili natančno zabeležen potek delovanja in ugotovitve za lažje nadaljnje odpravljanje napak oziroma programskih hroščev.

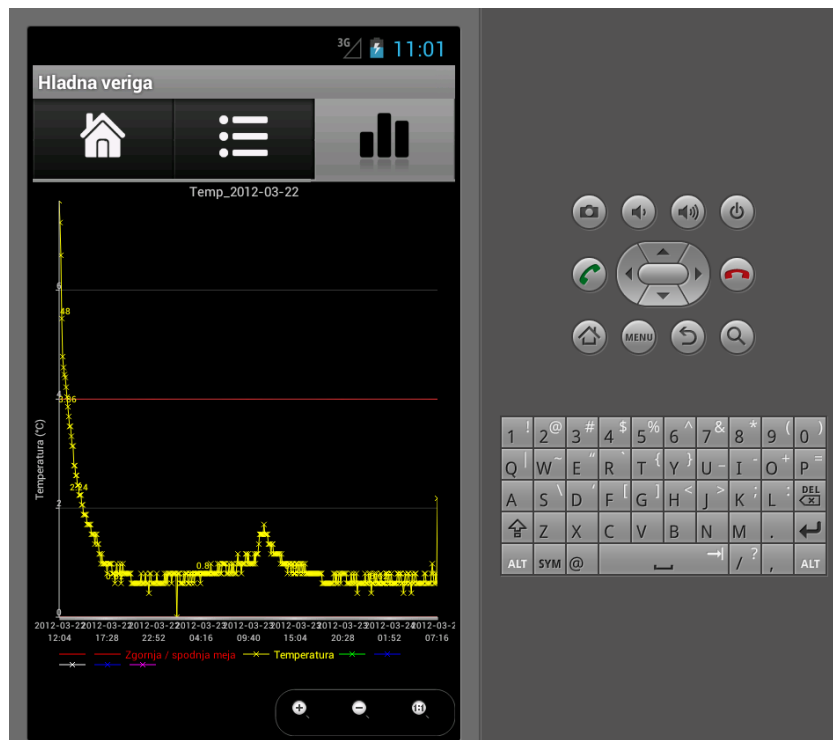
4.1 Testni primeri

Pripravili smo pet testnih primerov s katerim smo testirali aplikacijo, preverili pravilnost delovanja aplikacije in predstavili primere uporabe aplikacije:

- Test 1: zagon aplikacije, izbira datoteke, ki nima podatke o sledljivosti, izpis temperatur in izbira grafičnega prikaza.
- Test 2: zagon aplikacije, izbira datoteke, ki ima podatke o sledljivosti, izpis temperatur in izbira grafičnega prikaza.
- Test 3: zagon aplikacije, izbira datoteke, uporaba tipke nazaj, ponovna izbira datoteke, izpis temperatur in izbira grafičnega prikaza.
- Test 4: zagon aplikacije, izpis temperatur in izbira grafičnega prikaza.
- Test 5: zagon aplikacije, izbira datoteke, preklapljanje med zavihki.

4.2 Android Emulator

Android Emulator posnema mobilno napravo, ki nam omogoča razvoj in testiranje aplikacij brez uporabe fizične naprave (slika 23). Emulator posnema večino strojnih in programskih lastnosti tipičnega telefona, razen nekaterih, kot so opravljanje klicev, pošiljanje sporočil ter funkcij GPS modul. Možno je nastaviti različne konfiguracije naprav s pomočjo Android Virtual Device Manager. Virtualno napravo je možno zagnati kot samostojen program ali v razvojnem okolju Eclipse. Kljub pomanjkljivostim je mogoče takšne lastnosti simulirati. Klice ter pošiljanje lahko simuliramo med instancami emulatorjev skozi navidezno ustvarjeno omrežje, medtem ko GPS simulacije izvajamo s podajanjem geografskih točk skozi razvojno okolje.



Slika 23: Android Emulator.

4.3 Testni mobilnik

Za testiranje smo uporabili dva mobilna telefona:

- Telefon Wildfire S proizvajalca HTC ima kompaktno zaobljeno obliko in je toliko eleganten, kolikor je praktičen. Velikost je ravno pravšnja, da lepo sede v dlan, material ohišja pa je iz matirane plastike, ki posnema videz kovine. Od svojega predhodnika, modela Wildfire, se po obliki le minimalno razlikuje. Novi model je malo lažji in malo manjši, ima pa enako velik 3,2-palčni zaslon z malo višjo ločljivostjo 320 x 480 slikovnih točk.
- Telefon Samsung Nexus S deluje zelo kompaktno in elegantno. Zaslon je nekoliko ukrivljen, kar še izboljša uporabniško izkušnjo, s 129 grammi teže odlično leži v roki, velikost pa je še ravno pravšnja za enostavno delo z vsemi funkcijami. Pod pokrovom se skriva zmogljiva 1500 mAh baterija Li-ion, ki zagotavlja odlično avtonomijo mobilnika.

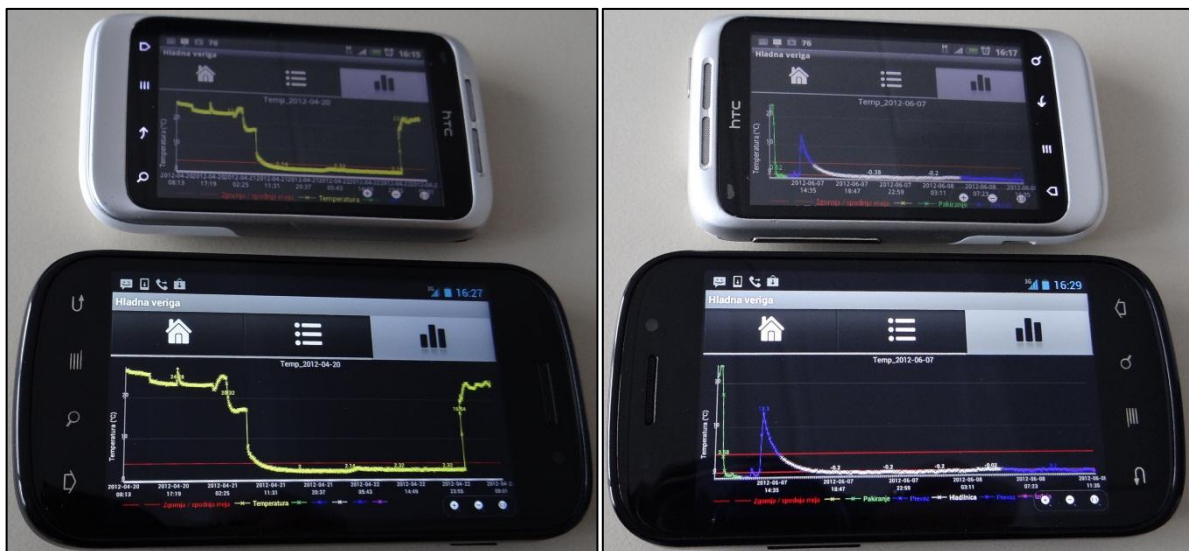
Tabela 3: Specifikacije mobilnega telefona [22].

	HTC Wildfire S	Samsung Nexus S
		
Procesor:	600 MHz ARM 11	1 GHz Cortex-A8
Spomin:	512 MB ROM, 512 MB RAM	512 MB RAM
Zaslon:	TFT kapacitivni zaslon na dotik, 256K barv. 320 x 480 pik, 8.13 cm	Super AMOLED kapacitivni zaslon na dotik ,16M barv. 480 x 800 pik, 10.16 cm
Baterija:	Li-Ion, kapacitete 1230 mAh	Li-Ion, kapacitete 1500 mAh
Omrežje:	GSM 850 / 900 / 1800 / 1900 HSDPA 900 / 2100	GSM 850 / 900 / 1800 / 1900 HSDPA 900 / 1700 / 2100
Platforma:	Android OS, v2.3 (Gingerbread)	Android OS, v2.3 (Gingerbread)
Kamera	5 MP, 2592 x 1944 pik, samodejno ostrenje, LED bliskavica	5 MP, 2560 x 1920 pik, samodejno ostrenje, LED bliskavica
Bluetooth:	Da, v3.0	Da, v2.1

4.4 Testiranje aplikacije

Aplikacijo smo najprej testirali na emulatorju Android 2.2, kjer smo poljubno in naključno izbrali vzorce testiranja. Pri tem nismo naleteli na težave pri delovanju aplikacije. Aplikacija se je odzivala po pričakovanjih, deluje stabilno. Hitrost delovanja je pričakovana, preklon med zasloni se odziva hitro, čakati moramo le pri nalaganju podatkov. V primeru da odpiramo datoteko, ki ima zapisano veliko število podatkov, se čas nalaganja poveča. Nato smo uporabili testne vzorce, ki smo jih pripravili. Tudi pri tem delu testiranja nismo naleteli na težave delovanja aplikacije. Aplikacijo smo nato testirali še na dveh mobilnikih. Na mobilniku po pričakovanjih aplikacija deluje hitreje in se bolj odziva, kot na emulatorju. Aplikacija deluje stabilno in se med izvajanjem ne zaključi samodejno zaradi kakšne napake. Pri poljubnem, oziroma naključnem testiranju aplikacije, ter pri testiranju s testnimi vzorci nismo naleteli na kakšne druge težave. Zaradi tehničnih razlik testnih mobilnih telefonov smo naleteli na dve razliki pri prikazu grafa:

- Hitrost delovanja aplikacije. Samsung Nexus S uporablja 1Ghz procesor in je v primerjavi s HTC-jevim 600 MHz procesorjem ARM 11 veliko hitrejši.
- Prikaz grafa. Zaradi velike razlike med resolucijo testnih mobilnih naprav (Samsung 480 x 800 pik, HTC 320 x 480 pik) je bila razlika pri velikosti črk na grafu, katerih se v uporabljeni knjižnici ne da povečati. Na sliki 24 je viden prikaz grafov na testnih mobilnih telefonih brez podatkov o sledljivosti (levo) in z podatki o sledljivosti (desno).

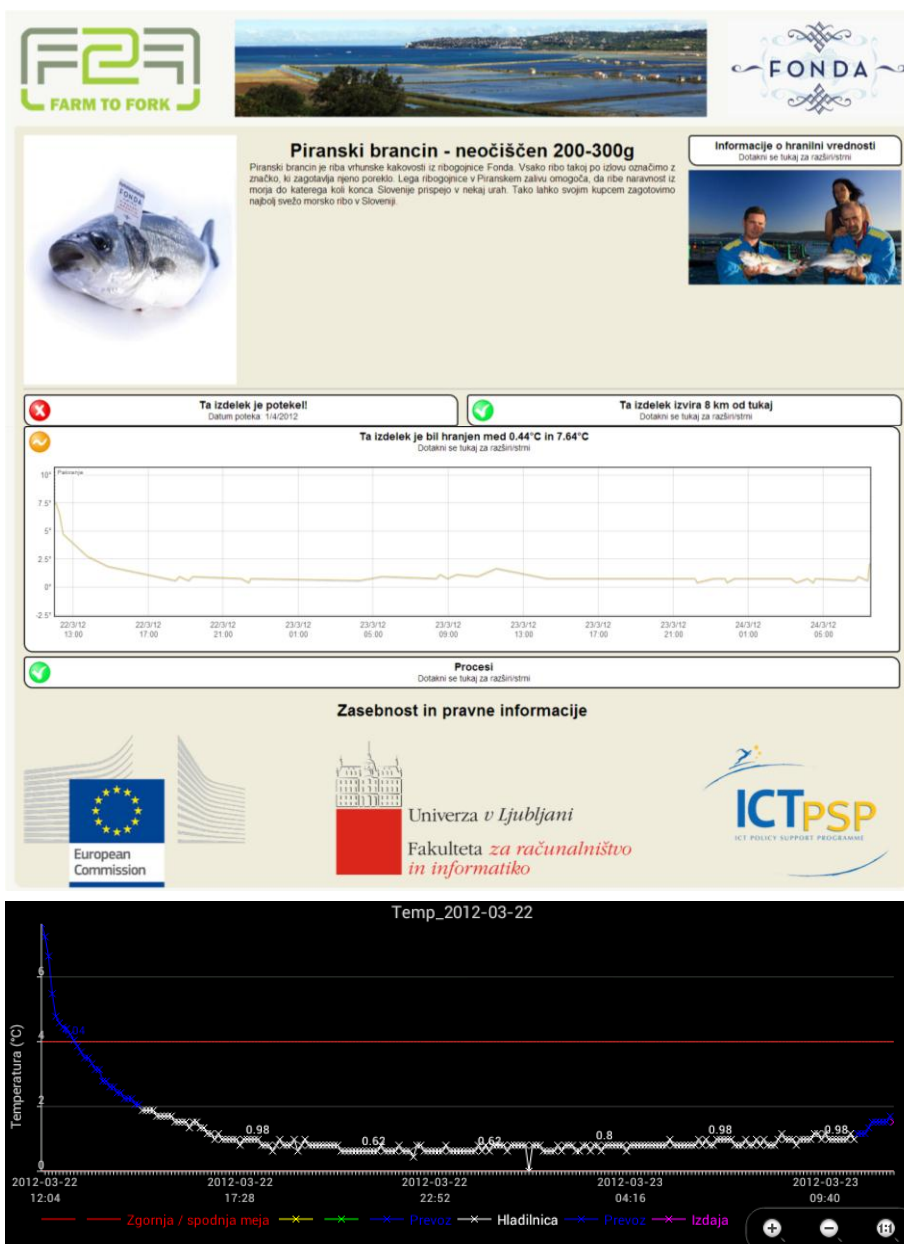


Slika 24: Prikaz grafov na testnih mobilnih telefonih.

Prikaz podatkov v grafičnem načinu smo tudi preverili z že izdelano spletno aplikacijo, ki je naložena na spletni strežnik Laboratorija za računalniško arhitekturo. To je projekt *RFID from Farm to Fork*, je potekal v štirih državah EU (Slovenija, Italija, Španija, Velika

Britanija) v letih 2010-2012. Realiziranih je pet pilotnih izvedb sistema sledljivosti za ribe, školjke, meso, sir in vino. Z uporabo radiofrekvenčne identifikacije (RFID) in v skladu z zakonodajo ter standardi označevanja proizvodov (GS1) je zagotovljeno pridobivanje, shranjevanje in dostop do podatkov, ki so na voljo vsem sodelujočim partnerjem v prehranski verigi, predvsem majhnim in srednje velikim podjetjem (SME).

Implementacija RFID sistema sledljivosti v Sloveniji je bila zasnovana in izpeljana na primeru piranskega brancina Fonda. Končna rešitev, poleg sistema sledljivosti, zajema tudi sistem hladne verige (ang. Cold Chain) ter spletno aplikacijo, ki potrošniku na mestu nakupa s skeniranjem QR kode (ang. Quick Response Code) omogoča dostop do podatkov o brancinu. Zajem in zbiranje podatkov o času, lokaciji in temperaturi poteka skozi celoten proces potovanja živila od ribogojnice do končnega kupca v ribarnici [23]. Slika 25 prikazuje graf prikazan na LRA strežniku in mobilni aplikaciji, izmerjeni dne 22.3.2012 do 24.3.2012.



Slika 25: Primerjava izrisa grafa iz spletne aplikacije z mobilno aplikacijo.

4.5 Namestitev aplikacije

Prenos aplikacije smo izvedli s pomočjo USB povezave računalnik – mobilni telefon. Iz namiznega računalnika smo na mobilni telefon preprosto prekopirali datoteko, ki se po prevajanju projekta generira v podmapi bin/. Po prenosu datoteke na pomnilniško kartico telefona, smo le tega varno odstranili iz računalnika.

Aplikacija se zažene z aktiviranjem datoteke na ikono, ki smo jo shranili na poljuben direktorij v mobilni napravi. Preden pa namestimo aplikacijo, moramo v nastavitvah mobilnega telefona vključiti opcijo za omogočanje namestitve datotek, ki niso prenesene iz Android trgovine, ampak so drugega izvora in so telefonu zato neznan. S tem se mobilni telefon zaščiti pred nepreverjenimi aplikacijami ter samodejnim nameščanjem le-teh. APK datotečni format je vrsta formata JAR. Uporablja se za distribucijo in namestitev komponent na mobilne platforme Android.

5 Zaključek

Mobilne platforme na pametnih mobilnih telefonih predstavljajo velik tržni delež mobilne telefonije in pridobivajo na vse večji popularnosti med uporabniki po celem svetu. Mobilna platforma Android je v zadnjem letu pridobila največ uporabnikov, saj se ne uporablja samo na mobilnih telefonih, temveč na vseh vrstah mobilnih naprav in tudi na televizijski platformi razvit s strani podjetja Google. Prihodnost razvoja lahko vidimo predvsem v odprto kodnih sistemih, ki pritegnejo zmeraj več ljudi in se razvijajo hitreje, ter prinašajo nove ideje.

V diplomskem delu smo spoznali pet mobilnih platform, podrobneje smo predstavili platformo Android in potek razvoja mobilne aplikacije. Razvijali in testirali smo mobilno aplikacijo v razvojnem okolju Eclipse, ki nam omogoča lažje in hitrejše delo. Razvili smo mobilno aplikacijo za prikaz senzorskih meritev, ki je informativne narave in omogoča prikaz temperatur. Na koncu smo mobilno aplikacijo testirali na mobilnem telefonu HTC Wildfire S in Samsung Nexus S, ki je zaradi boljše procesorske moči deloval bolje in hitreje.

Razvoj s pomočjo razvojnega okolja Eclipse se zdi precej enostaven in fleksibilen, na spletu pa je možno najti veliko tehničnih informacij s katerimi je programiranje v programskem jeziku Java na platformah Android precej lažje. Internetne informacije in množice primerov kod so nam bili v veliko pomoč tudi pri samem razvoju aplikacije.

DODATEK: SEZNAM SLIKM IN TABEL

Slika 1: Tipi mobilnih naprav.....	6
Slika 2: Uporaba pametnih telefonov.....	7
Slika 3: Tržni delež mobilnih operacijskih sistemov za leto 2012 [9].....	8
Slika 4: iOS logotip.....	9
Slika 5: Windows Mobile logotip.....	10
Slika 6: BlackBerry logotip.....	11
Slika 7: Symbian logotip.....	11
Slika 8: Android logotip.....	12
Slika 9: Android arhitektura [17].....	15
Slika 10: Razvojno okolje Eclipse.....	17
Slika 11: Podatki o datotekah.....	19
Slika 12: Podatki o temperaturah.....	20
Slika 13: Podatki o sledljivosti.....	20
Slika 14: Shema funkcionalnega delovanja aplikacije.....	23
Slika 15: Vstopno okno.....	24
Slika 16: Povzetek datoteke.....	25
Slika 17: Izpis temperatur.....	26
Slika 18: Datotečna struktura.....	26
Slika 19: Izris grafa brez podatkov o sledljivosti.....	27
Slika 20: Izris grafa z podatki o sledljivosti.....	28
Slika 21: Logotip aplikacije.....	29
Slika 22: Grafika za gumbe.....	29
Slika 23: Android Emulator.....	31
Slika 24: Prikaz grafov na testnih mobilnih telefonih.....	33
Slika 25: Primerjava izrisa grafa iz spletne aplikacije z mobilno aplikacijo.....	34
Tabela 1: Operacijski sistem Windows Mobile.....	10
Tabela 2: Android različice.....	13
Tabela 3: Specifikacije mobilnega telefona [22].....	32

Literatura

- [1] (2012) Smartphone statistic and market share. Dostopno na:
<http://www.email-marketing-reports.com/wireless-mobile/smartphone-statistics.htm>
- [2] (2012) Google Play. Dostopno na:
<http://developer.android.com/google/play-services/index.html>
- [3] (2012) Prva generacija mobilnih komunikacij. Dostopno na:
<http://en.wikipedia.org/wiki/1G>
- [4] (2012) Druga generacija mobilnih komunikacij. Dostopno na:
<http://en.wikipedia.org/wiki/2G>
- [5] (2012) Tretja generacija mobilnih komunikacij. Dostopno na:
<http://en.wikipedia.org/wiki/3G>
- [6] (2012) Četrta generacija mobilnih komunikacij. Dostopno na:
<http://en.wikipedia.org/wiki/4G>
- [7] (2012) Mobilni telefoni. Dostopno na:
<http://www.telegraph.co.uk/technology/mobile-phones/#>
- [8] (2012) Android App Market. Dostopno na:
<http://www.android-app-market.com/android-architecture.html>
- [9] (2012) SRC. Dostopno na:
<http://infosrc.wordpress.com/2012/09/27/centralizirano-upravljanje-mobilnih-naprav-ii-del/>
- [10] (2012) Operacijski sistem iOS. Dostopno na:
<http://www.apple.com/ios/>
- [11] (2012) Operacijski sistem Windows Mobile. Dostopno na:
<http://gizmodo.com/5374876/windows-mobile-65-review-theres-no-excuse-for-this>
- [12] (2012) Operacijski sistem BlackBerry. Dostopno na:
<http://us.blackberry.com/software/smartphones/blackberry-6-os.html>
- [13] (2012) Operacijski sistem Symbian. Dostopno na:
<http://www.symbianos.org/intro>
- [14] (2012) Operacijski sistem Android. Dostopno na:
<http://www.android.com/about/>
- [16] (2012) Različice operacijskega sistema Android. Dostopno na:
<http://www.techgainer.com/the-android-story-infographic/>
- [19] (2012) Eclipse. Dostopno na:
<http://www.eclipse.org/org/>
- [20] (2012) XML. Dostopno na:
http://www.w3schools.com/xml/xml_what_is.asp
- [21] (2012) Android SDK. Dostopno na:
<http://developer.android.com/sdk/index.html>

- [22] (2012) GSMARENA. Dostopno na:
<http://www.gsmarena.com/compare.php3?idPhone1=3777&idPhone2=3620>
- [23] (2012) Laboratorij za računalniško arhitekturo. Dostopno na:
<http://ira.fri.uni-lj.si/slo/projekti/rfid-f2f/>