

Šolski center Celje
Srednja šola za kemijo, elektrotehniko in računalništvo

ŠOLSKA ANDROID APLIKACIJA

raziskovalna naloga

Avtorji:

Luka FISTRIĆ, E-4. e
Luka ŠTOREK, E-4. e
Darko VOZLIČ, E-4. e

Mentor:

Mojmir KLOVAR univ. dipl. inž

Mestna občina Celje, Mladi za Celje

Celje, 2012

KAZALO

1 POVZETEK	2
1.1 UVOD	3
2 IZVEDBA NAČRTA ANDROID APLIKACIJE	4
2.1 Razvojno okolje aplikacije.....	4
2.2 Vremenski modul.....	4
2.3 Redovalnica	5
2.4 Nadomeščanja	6
3 ZAKLJUČEK	7
4 VIRI IN LITERATURA	8
5 ZAHVALA.....	9
6 PRILOGE	10
6.1 Izbirni meni aplikacije	10
6.2 Vreme	10

1 POVZETEK

Pridobivanje informacij in shranjevanje podatkov ima v današnji družbi zelo velik pomen, saj si praktično ne moremo več predstavljati, da ne bi mogli slediti najnovejšim spremembam v svetu. Zato je potrebno uporabnikom raznoraznih storitev omogočiti čim hitrejšo pridobivanje informacij v trenutku, ko jih ta želi pregledovati.

Prav zaradi tega razloga smo se lotili raziskovanja v tej nalogi, kako pridobiti podatke v najkrajšem možnem času in jih shranjevati, da so uporabniku na voljo skozi celoten čas delovanja aplikacije, ob tem pa je zagotovljena tudi njihova varnost.

1.1 UVOD

V današnjem času obstaja mnogo načinov, kako prenašati podatke med posameznimi napravami in vzpostaviti interakcijo med njimi. Za kompatibilnost med temi pa je potrebno poskrbeti s pomočjo programske kode in prav to je glavni razlog, da smo se lotili te raziskovalne naloge ter se soočili s programiranjem aplikacije za Android operacijski sistem 2.3.3 verzije, ki je trenutno najbolj razširjena med napravami z Android OS. S to aplikacijo želimo končnemu uporabniku zagotoviti karseda preprost uporabniški vmesnik, da se bo lahko hitro privadil na samo aplikacijo in si z njeno uporabo pohitril ter olajšal pridobivanje informacij, povezanih s šolo. Z njo lahko na enostaven način preverja vremensko stanje v Celju in spremlja šolska obvestila ter pomembne dogodke. Ob tem lahko na enostaven način preverja urnike ter pregleduje vozni red vlakov na želeni relaciji. Osebna evidenca ocen pa mu omogoča sprotno vnašanje teh in sprotno izračunavanje povprečja.

Za glavni cilj raziskave smo si zastavili problem, kako zagotoviti usklajenost med podatki, ki jih pridobivamo iz različnih virov in jih prikazujemo v aplikaciji oziroma njenih pripomočkah. S tem je potrebno uporabniku omogočiti berljiv način podatkov. Pomemben del naše raziskave je tudi zaščita podatkov, s katero bomo med samimi podatki in uporabnikom vzpostavili neke vrste kontrolo s pomočjo ključa, brez katerega bo dostop do podatkov onemogočen. Na ta način bo varnost aplikacije močno povečana.

2 IZVEDBA NAČRTA ANDROID APLIKACIJE

Na samem začetku projekta smo v skupini združili vse ideje o tem, kaj bi aplikacija omogočala in ugotovili, da bi bilo pametno uporabiti podatke iz vremenske postaje na Šolskem centru Celje in ob tem še druge pripomočke za hiter dostop do šolskih informacij ter voznih redov. Pred izvedbo načrtanega postopka smo si med sabo porazdelili programiranje posameznih delov aplikacije tako, da smo si omogočili čim enostavnejši in hitrejši proces programiranja. Ob porajanju kakšnega vprašanja ali problema smo eden drugemu priskočili na pomoč in skupaj razrešili dano situacijo. Ravno v tem se kaže čar timskega dela več ljudi, ki temelji na medsebojni pomoči in izmenjevanju ter deljenju znanja posameznikov.

2. 1 Razvojno okolje aplikacije

Razvoj spletne aplikacije je potekal v programskem okolju Eclipse IDE, ki omogoča, da lahko z nameščenim Android SDK-jem zaženete Android aplikacijo tudi v primeru, da si ne lastite nobenega izmed mobilnih telefonov z Android operacijskim sistemom. Programska koda za izvajanje je zapisana v programskem jeziku Java. Kot je bilo že prej omenjeno je za programiranje in delovanje programa v Android OS potrebno namestiti dodaten SDK, imenovan Android SDK in nato še ADT Plugin. Ti dve dodatni orodji pripomoreta k temu, da lahko aplikacijo poženete kar v tako imenovanem "Emulatorju", ki simulira mobilni telefon z nameščenim Android OS-jem kar na vašem računalniku. Prav tako pa si morate namestiti tudi verzijo Androida, v kateri želite programirati in si jo preprosto izberete iz Android SDK-ja.

2. 2 Vremenski modul

Pri vremenskem modulu je bilo potrebno prebrati podatke s spletne strani, kjer so vneseni, zapisani v .txt formatu ter ločeni s podpičjem. Najprej smo se odločili kdaj bomo naložili podatke. To smo izpeljali tako, da v kolikor aplikacija ne dobi odziva spletne strani, kar lahko povzroči, da telefon ni povezan na internet, naloži zadnje podatke, ki so se zapisali ob zadnji posodobitvi v tekstovno datoteko na mobitelu. Vsakič ko se podatki lahko posodobijo, se le ti tudi shranijo v datoteko. Da smo lahko dostopali do naslova in brali podatke s spleta, smo morali uporabiti metodo z HTTPGet elementi. Kjer je metoda kot vhodni parameter dobila url, kjer se nahajajo podatki ter dobila odziv od spletne strani. Druga metoda pa je prebrala vsebino, ki je na strani. Ta koda ni brala izvorne kode strani ampak samo vsebino na strani, ker je format podatkov samo vrstica s tekstom, se je celotna vrstica

prebrala v eno spremenljivko, v kolikor bi se dogodilo, da so podatki v več spremenljivkah, je metoda prilagojena, da preveri kdaj je konec dokumenta. Ko smo podatke pridobili iz spleta, so bili sestavljeni v eni spremenljivki. Za ločevanje teh podatkov smo uporabili že obstoječe metode za rezanje teksta, uporabili smo metodo split in ker so podatki v sledeči obliki `podatek1;podatek2;podatek3` smo nastavili da vsak podatek »odreže« ko naleti na dvopičje. Vse podatke ki jih pridobimo s spleta shranjujemo najprej v polje stringov (nizov) ter jih nato prikazujemo v aplikaciji. Zaradi striktnega načina programiranja, ki v Android aplikacijah velja od 3.0 verzije, smo morali vse klice izvajati v posebnih metodah, saj javni klici na novejših napravah ne delujejo več. Uporabniku je omogočeno, da podatke osveži kadar sam želi, s pritiskom na sliko osveži, podatki pa se tudi sami posodobijo ob odprtju forme vreme.

2.3 Redovalnica

Ob začetku projekta smo si zastavili, da mora redovalnica omogočati dodajanje poljubnih predmetov in ocen pri vnesenih predmetih ter tudi naknadno dodajanje in morebitno popravljanje. Sproti izračunava tudi povprečje skozi celotno obdobje danega predmeta. Uporabniku omogoča dodajanje predmetov, glede na njegove potrebe. To pomeni, da je neodvisna, saj si uporabnik sam vnese svoje šolske predmete. Najprej sem v .xml datoteki dodelal dizajn te naše evidence ocen tako, da sem uporabil elemente kot je `LinearLayout`, ki ga uporabljamo za postavitve aplikacijskih elementov v njem kot so na primer razni gumbi in elementi za vnašanje ter prikazovanje podatkov. V to postavitev sem dodal še gumb, s katerim uporabnik dodaja predmete in se s pritiskom nanj prikaže virtualna tipkovnica na zaslonu, ki omogoča uporabniku vnašanje zelenih predmetov. Za prikazovanje predmetov in pripadajočih ocen sem uporabil `ListView` komponento, ki služi ravno takšni namembnosti kot je ta, da prikazujemo zelene podatke.

Za tem je bilo potrebno v Java programskem jeziku sprogramirati medsebojno delovanje zgoraj opisanih komponent. Da lahko z elementi iz postavitve sploh upravljamo, jih je potrebno najprej deklarirati in se sklicevati na id elementa iz prej dodelane postavitve. Nato sem sprogramiral metodo za pridobivanje vnesenih podatkov, ki sem jo zatem klical v drugi metodi, sprogramirani za odzivanje glede na klik gumba. Ta metoda ob sprožitvi prikaže tipkovnico za vnašanje podatkov. Za ohranjanje podatkov tudi po zaprti aplikaciji, sem v eni izmed metod omogočil shranjevanje teh, saj bi bilo vnašanje ob vsakem zagonu aplikacije brezsmiselno in ne bi služilo osnovnemu namenu.

2. 4 Nadomeščanja

Moja naloga je bila, da iz spletne strani Srednje šole KER preberem podatke in jih prikažem na mobilni napravi tako, da si bo vsak dijak lahko ogledal nadomeščanja, ki so objavljena za njegov razred. Cilj je bil, da bi bila ta stran čim bolj preprosta oz. po možnosti takšna, ki jo dijaki že dobro poznajo.

Naloge sem se lotil postopoma in sem si najprej ogledal zgradbo spletne strani, iz katere naj bi podatke pridobival. Zatem sem si ogledal na kakšne načine je mogoče ta problem rešiti in sem naletel na zelo preprosto in pregledno rešitev. Rešitev je bila v WebView-u, ki podobno kot iFrame v HTML kaže na drugo stran, oz. v tem primeru je to program, ki stran pokaže takšno kot jo vidite v brskalniku. Zadeva se je na prvi pogled zdela dokaj preprosta in sem kmalu imel izdelek pripravljen. Med testiranjem pa sem se vprašal, kaj bi bilo, če dijak slučajno ne bi imel dostopa do interneta. Tukaj pa se je vse obrnilo na glavo, ampak čez nekaj časa sem spet prišel do rešitve in tako sem program priredil tako, da se stran shranjuje v pomnilnik in se, v primeru da nimamo povezave, prikažejo zadnji shranjeni podatki iz strani. Do te rešitve sem prišel z »HttpClient.execute()« in stran shranjeval v »[file://<lokacija dokumenta>](#)«. Na koncu sem vse skupaj prenesel v naš osnovni obrazec in s tem delom uspešno zaključil.

3 ZAKLJUČEK

Pri programiranju aplikacij lahko dani problem rešimo na več načinov tako, da imamo svobodno izbiro pri pisanju programske kode, vendar je potrebno biti vseeno previden pri pomenu raznih ukazov, ki morajo biti pravilni sintaktično in semantično.

Na zahtevno stopnjo težavnosti pri programiranju naše aplikacije smo naleteli pri pridobivanju informacij iz vremenske postaje, saj je bilo potrebno podatke, ločene s podpičjem, prebrati in shraniti v pripadajoče spremenljivke. Na začetku je problem nastal tudi pri zaščiti ter shranjevanju podatkov ostalih pripomočkov aplikacije, saj je bilo potrebno zagotoviti obstojnost podatkov po zaprti aplikaciji.

Izdelava te aplikacije v skupini je bil naš prvi projekt take vrste in se je tudi zelo dobro obnesel. Z delom take vrste smo se na ta način že privadili na bodoče delo, ki bo potekalo v kakšnem izmed podjetij in bo potrebno medsebojno združevati ter deliti znanje. Pri izdelavi smo se prvič soočili s programskim jezikom Java, saj v šoli uporabljamo C#, vendar sta si jezika med sabo precej podobna tako, da večjih težav pri programiranju ni bilo.

4 VIRI IN LITERATURA

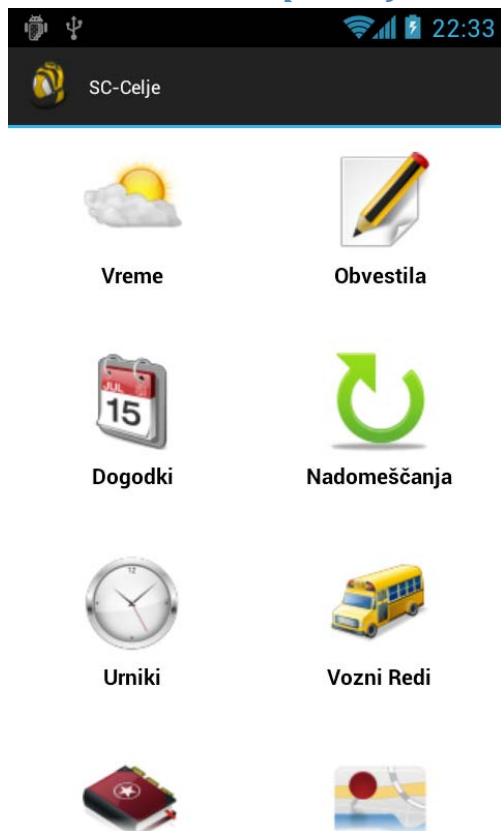
- M. Gargenta (2011): Learning Android. Sebastopol: O'Reilly Media.
- Spletna stran Android razvijalcev
<http://developer.android.com/index.html>

5 ZAHVALA

Zahvala pri izpeljevanju načrta zamišljene aplikacije gre g. Mojmirju Klovarju, ki nas je skozi celoten cikel raziskovanja spodbujal in nam pomagal razreševati težave pri nastajanju našega končnega izdelka.

6 PRILOGE

6.1 Izbirni meni aplikacije



6.2 Vreme

