

1

Osnove Android programiranja

ŠTA ĆETE NAUČITI U OVOM POGLAVLJU

- ▶ Šta je Android?
- ▶ Različite Android verzije i skup specifičnih funkcija
- ▶ Android arhitektura
- ▶ Različiti Android uređaji prisutni na tržištu
- ▶ Android Market prodavnica aplikacija
- ▶ Kako da pronađete alate i SDK za razvoj Android aplikacija?
- ▶ Kako da razvijete svoju prvu Android aplikaciju?

Dobrodošli u svet Android operativnog sistema! Kada sam počeo da pišem svoju prvu knjigu o Android platformi (što je bilo pre manje od godinu dana), istakao sam da se Android nalazi na drugom mestu na tržištu pametnih telefona u SAD, odmah iza BlackBerry platforme kompanije „Research In Motion“ (RIM), a ispred iPhone platforme kompanije „Apple“. Ubrzo nakon objavljivanja knjige, comScore (globalni lider u ocenjivanju digitalnog sveta i preporučeni izvor podataka koji se odnose na istraživanje digitalnog tržišta) objavio je da je Android pretekao BlackBerry i postao najpopularnija platforma za pametne telefone u SAD.

Nekoliko meseci kasnije, kompanija „Google“ je izdala Android 3.0, sa kodnim nazivom *Honeycomb*. U Android 3.0 verziji „Google“ je u skupu razvojnih alata (Software Development Kit) uveo nekoliko novih funkcija, koje su predviđene za uređaje sa širokim ekranom, posebno za tablet uređaje. Ukoliko razvijate aplikacije za Android pametne telefone, Android 3.0 nije preterano koristan, zato što nove funkcije nisu podržane na pametnim telefonima. Istovremeno sa

izdavanjem Android 3.0 verzije, „Google“ je počeo razvoj nove verzije Android platforme, koja se može koristiti podjednako dobro i na pametnim telefonima i na tablet računarima. U oktobru 2011. godine „Google“ je izdao Android 4.0, sa kodnim nazivom Ice Cream *Sandwich*, koji je opisan u ovoj knjizi.

U ovom poglavlju ćete naučiti šta je Android i šta ga čini podjednako privlačnim i programerima i proizvođačima uređaja. Takođe ćete pristupiti razvoju svoje prve Android aplikacije, odnosno saznaćete kako da pronađete sve neophodne alate i kako da ih ispravno podesite da biste mogli da testirate svoju aplikaciju na Android 4.0 emulatoru. Nakon što pročitate ovo poglavlje, imaćete osnovno znanje koje je neophodno da pristupite proučavanju složenijih tehnika i trikova koji se odnose na razvoj atraktivnih Android aplikacija.

ŠTA JE ANDROID?

Android je mobilni operativni sistem koji je zasnovan na modifikovanoj verziji Linux operativnog sistema. Originalno ga je razvila „Android, Inc.“ kompanija. U toku 2005. godine, kao deo strategije za pristupanje tržištu mobilnih uređaja, „Google“ je kupio Android i preuzeo odgovornost za njegov dalji razvoj (kao i za razvojni tim).

Želeo je da Android bude otvoren i slobodan; zbog toga je najveći deo Android koda bio pod Apache License licencom, koja se odnosi na otvoreni kod, što znači da svako može da koristi Android preuzmanjem njegovog izvornog koda. Štaviše, proizvođači (pre svega, hardvera) mogu da dodaju sopstvene ekstenzije i podešavaju Android u skladu sa svojim proizvodima kako bi se oni razlikovali u odnosu na druge proizvode. Ovaj jednostavan model čini Android veoma atraktivnim i zbog toga je privukao brojne proizvođače. Ovo je posebno značajno za kompanije koje su bile pogođene fenomenom iPhone telefona kompanije „Apple“, neverovatno uspešnim proizvodom koji je izazvao revoluciju u industriji pametnih telefona. Takve kompanije su, između ostalog, „Motorola“ i „Sony Ericsson“, koje su dugo godina razvijale sopstvene operativne sisteme za mobilne uređaje. Nakon što je iPhone pušten u prodaju, neki od proizvođača su morali da pronađu način da revitalizuju svoje proizvode. Oni su kao rešenje videli Android – nastaviće razvoj sopstvenih hardvera i koristiti Android kao operativni sistem koji upravlja tim hardverom.

Glavna prednost prilagođavanja Android operativnog sistema je primena unifikovanog pristupa razvoju aplikacija. Programeri razvijaju sve pod Android platformom, a njihove aplikacije treba da imaju mogućnost izvršavanja na velikom broju različitih uređaja, pod uslovom da ti uređaji koriste Android operativni sistem. U svetu pametnih telefona aplikacije su najznačajniji deo lanca uspeha. Proizvođačima uređaja je Android platforma, zbog toga, bila najbolja osnova za „hvatanje u koštac“ sa iPhone telefonom, čiji su korisnici već imali veliki broj aplikacija na raspolaganju.

Android verzije

Android je doživeo veliki broj modifikacija od svoje prve verzije. U tabeli 1-1 prikazane su različite Android verzije sa njihovim kodnim nazivima.

TABELA 1-1 Kratak pregled Android verzija

Android verzija	Datum objavljivanja	Kodni naziv
1.1	9. februar 2009.	
1.5	30. april 2009.	Cupcake
1.6	15. septembar 2009.	Donut
2.0/2.1	26. oktobar 2009.	Eclair
2.2	20. maj 2010.	Froyo
2.3	6. decembar 2010.	Gingerbread
3.0/3.1/3.2	22. februar 2011.	Honeycomb
4.0	19. oktobar 2011.	Ice Cream Sandwich

U februaru 2011. godine „Google“ je izdao Android 3.0 verziju koja je prilagođena isključivo za tablet računare sa širokim ekranom.

Ključne promene koje je uveo Android 3.0 su navedene u daljem tekstu.

- 3D desktop sa novim dodacima
- poboljšani multi-tasking
- nove funkcije web čitača, kao što su korišćenje kartica, automatsko popunjavanje formi, sinhronizacija označenih stranica i privatnost prilikom pregledanja
- podrška za procesore sa više jezgara

Aplikacije koje su napisane za Android verzije pre 3.0 su kompatibilne sa Android 3.0 uređajima i izvršavaju se bez ikakvih modifikacija. Međutim, Android 3.0 tablet aplikacije, koje mogu da koriste najnovije funkcije raspoložive u 3.0 verziji, ne mogu da se izvršavaju na starijim uređajima. Da biste se uverili da Android tablet aplikacija može da se izvršava na svim verzijama uređaja, morate programski da obezbedite da se određene funkcije koriste samo na onim uređajima na kojima se izvršavaju specifične Android verzije.

U oktobru 2011. godine „Google“ je izdao Android 4.0, verziju koja je omogućila korišćenje svih funkcija uvedenih u Android 3.0 verziji na pametnim telefonima, zajedno sa nekim novim funkcijama, kao što su otključavanje na osnovu prepoznavanja lica, nadgledanje i upravljanje korišćenjem podataka, Near Field Communication (NFC) i drugo.

Android karakteristike

Pošto Android ima otvoreni kod i dostupan je potpuno besplatno proizvođačima za samostalno prilagođavanje, ne postoje precizno definisane hardverske ili softverske konfiguracije. Međutim, Android podržava sledeće:

- skladištenje – Za skladištenje podataka koristi se SQLite, jednostavna relacionalna baza podataka. Poglavlje 6 sadrži detaljan opis postupaka skladištenja podataka.
- pristupanje mrežama – Podržane su GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth (uključuje A2DP i AVRCP), Wi-Fi, LTE i WiMAX. U Poglavlju 8 detaljno su opisani postupci umrežavanja.

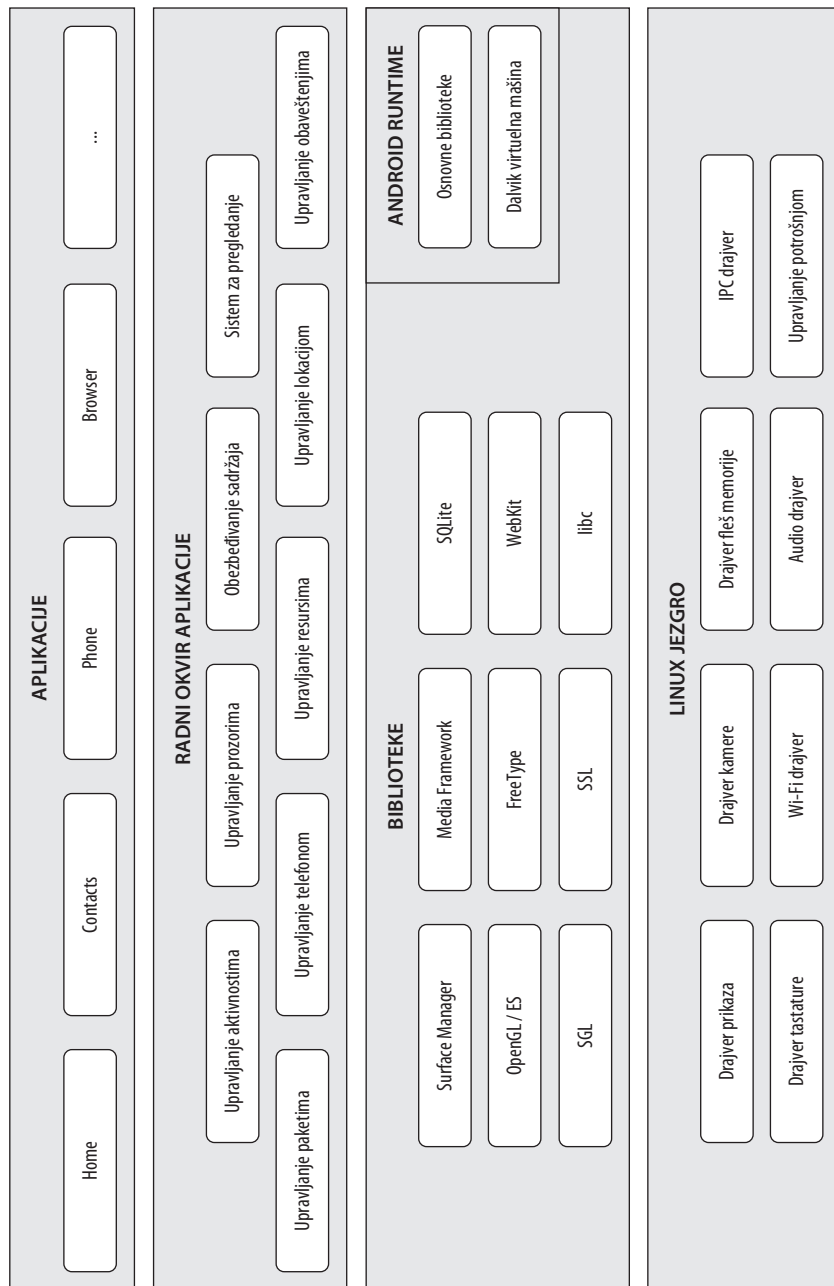
- slanje poruka – Postoji podrška za SMS i MMS. U Poglavlju 8 detaljno je opisan postupak slanja poruka.
- web čitač – Zasnovan je na WebKit otvorenom kodu, kao i na V8 JavaScript okruženju koje koristi Chrome.
- multimedijalna podrška – Postoji podrška za sledeće formate medija: H.263, H.264 (u 3GP ili MP4 kontejneru), MPEG-4 SP, AMR, AMR-WB (u 3GP kontejneru), AAC, HE-AAC (u MP4 ili 3GP kontejneru), MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF i BMP
- hardverska podrška – Akcelerometrijski senzor, kamera, digitalni kompas, senzor rastojanja i GPS
- multi-touch – Podrška za multi-touch ekrane
- multi-tasking – Podrška za aplikacije koje izvršavaju više zadataka istovremeno
- flash podrška - Android 2.3 podržava Flash 10.1.
- povezivanje – Postoji podrška za deljenje internet konekcija i žičnih/bežičnih pristupnih tačaka.

Android arhitektura

Da biste razumeli na koji način funkcioniše Android, pogledajte sliku 1-1, na kojoj su prikazani različiti slojevi koji sačinjavaju Android operativni sistem (OS).

Android OS je grubo podeljen na pet sekcija i četiri osnovna sloja:

- Linux jezgro – To je jezgro na kome je Android zasnovan. Ovaj sloj sadrži sve drajvere uređaja definisane na niskom nivou za različite hardverske komponente svakog pojedinačnog Android uređaja.
- biblioteke – Sadrže sav kod koji obezbeđuje osnovne funkcije Android operativnog sistema. Na primer, SQLite biblioteka obezbeđuje podršku za korišćenje baza podataka, tako da aplikacija može da je koristi za skladištenje podataka. WebKit biblioteka obezbeđuje funkcije koje se odnose na pregledanje web sadržaja.
- Android runtime okruženje – Na istom nivou kao i biblioteke, Android okruženje obezbeđuje skup osnovnih biblioteka koje omogućavaju programerima da pišu Android aplikacije korišćenjem Java programskog jezika. Android okruženje sadrži i Dalvik virtuelnu mašinu, koja omogućava svakoj Android aplikaciji da se izvršava u sopstvenom procesu, sa sopstvenom instancom Dalvik virtuelne mašine (Android aplikacije se prevode u Dalvik izvršne datoteke). Dalvik je specijalizovana virtuelna mašina, projektovana specijalno za Android i optimizovana za mobilne uređaje koji koriste baterije pri radu i imaju ograničene memorijske resurse i CPU (centralnu procesorsku jedinicu).
- radni okvir aplikacija – Omogućava se korišćenje različitih mogućnosti Android operativnog sistema, tako da programeri mogu da ih koriste u svojim aplikacijama.
- aplikacije – Na ovom nivou nalaze se aplikacije koje se isporučuju sa Android uređajima (kao što su Phone, Contacts, Browser i slične), kao i aplikacije koje se preuzimaju i instaliraju korišćenjem Android Marketa. Sve aplikacije koje samostalno kreirate nalaze se na ovom nivou.



SLIKA 1-1

Android uređaji na tržištu

Android uređaji postoje u različitim oblicima i veličinama. Od kraja novembra 2011. godine Android operativni sistem se koristi na sledećim tipovima uređaja:

- pametni telefoni (*eng.* Smartphones)
- tablet računari (*eng.* Tablets)
- uređaji za čitanje elektronskih knjiga (*eng.* E-reader devices)
- netbuk računari (*eng.* Netbooks)
- MP4 plejeri
- internet TV uređaji

Velika je verovatnoća da imate bar jedan od prethodno navedenih uređaja. Na slici 1-2 (gledano sleva udesno) prikazani su Samsung Galaxy S II, Motorola Atrix 4G i HTC EVO 4G pametni telefoni.



SLIKA 1-2

Još jedna popularna kategorija uređaja koje proizvođači ubrzano izbacuju na tržište su *tablet* računari. Oni se, uglavnom, prave u dve dimenzije: sedam inča i deset inča, mereno dijagonalno. Na slici 1-3 prikazani su Samsung Galaxy Tab 10.1 (levo) i Asus Eee Pad Transformer TF101 (desno), a oba su tableti veličine 10,1 inč. I Samsung Galaxy 10.1 i Asus Eee Pad Transfer TF101 koriste Android 3 operativni sistem.

**SLIKA 1-3**

Android počinje da se koristi i na posebnim uređajima, kao što su uređaji za čitanje elektronskih knjiga. Na slici 1-4 prikazani su NOOK Color kompanije „Barnes and Noble“ (levo) i Kindle Fire kompanije „Amazon“ (desno), koji su primer kolor uređaja za čitanje elektronskih knjiga koji koriste Android operativni sistem.

**SLIKA 1-4**

Android polako pronalazi svoje mesto i u našim domovima. Zaposleni u švedskoj kompaniji „Lava“ su razvili TV zasnovan na Android operativnom sistemu, koji su nazvali Scandinavia Android TV (videti sliku 1-5).

„Google“ je razvio specifičnu platformu za pametne televizore, koja je zasnovana na Android operativnom sistemu, i to u saradnji sa kompanijama kao što su „Intel“, „Sony“ i „Logitech“. Na slici 1-6 prikazan je Google TV, koji je kreirala „Sony“ kompanija.



SLIKA 1-5



SLIKA 1-6

U vreme pisanja ove knjige Samsung Galaxy Nexus (videti sliku 1-7) bio je jedini uređaj koji je koristio Android 4.0 operativni sistem. Međutim, iz kompanije „Google“ stiglo je obećanje da će svi postojeći uređaji (kao što je Nexus S) imati mogućnost prelaska na Android 4.0. Kada budete čitali ovu knjigu, verovatno će postojati veliki broj uređaja na kojima se izvršava Android 4.0.



SLIKA 1-7

Android Market

Kao što je prethodno pomenuto, jedan od glavnih faktora za utvrđivanje uspeha određene platforme pametnih telefona je postojanje aplikacija koje je podržavaju. Jasno je, na osnovu uspeha iPhone telefona, da aplikacije imaju veoma značajnu ulogu u utvrđivanju uspeha ili neuspeha nove platforme. Pored toga, mogućnost prosečnih korisnika da pristupaju ovim aplikacijama je izuzetno značajna.

Zbog toga je „Google“ u avgustu 2008. godine najavio Android Market, mrežno dostupnu prodavnicu aplikacija za Android uređaje, a pristupanje je omogućio korisnicima dva meseca kasnije. Koristeći Market aplikaciju koja je preinstalirana na Android uređaju, korisnici su mogli veoma

jednostavno da preuzimaju aplikacije nezavisnih autora direktno na svoje uređaje. Android Market je predviđen i za besplatne aplikacije i za one koje se plaćaju. Aplikacije koje se plaćaju dostupne su samo u određenim zemljama, zbog pravnih ograničenja. Slično tome, u nekim zemljama aplikacije sa Android Marketa korisnici mogu kupovati, ali programeri ih ne mogu prodavati. Na primer, u vreme pisanja ove knjige korisnici u Indiji su mogli da kupuju aplikacije sa Android Marketa, ali programeri u toj zemlji nisu mogli da prodaju aplikacije na Android Marketu. Postoje i obrnute situacije; na primer, u Južnoj Koreji korisnici ne mogu da kupuju aplikacije, ali programeri mogu da prodaju aplikacije na Android Marketu.



NAPOMENA Poglavlje 12 sadrži više informacija o Android Marketu i načinu na koji možete da prodajete svoje aplikacije na njemu.

Android programerska zajednica

Posle četiri verzije Android operativnog sistema, razvijena je velika zajednica programera širom sveta. Sada je mnogo jednostavnije da dobijete rešenje određenih problema, odnosno da pronađete srodne programere sa kojima možete da podelite ideje o aplikacijama i da razmenjujete iskustva.

Postoje određene programerske zajednice/stranice koje možete da posetite da biste potražili pomoć ukoliko se pojave određeni problemi pri radu sa Android operativnim sistemom:

- **Stack Overflow** (www.stackoverflow.com) - Stack Overflow je stranica sa pitanjima i odgovorima, koju kolaborativno razvijaju programeri. Ukoliko imate neko pitanje koje se odnosi na Android operativni sistem, velika je verovatnoća da ćete na Stack Overflow sajtu pronaći diskusije o istoj temi i da je neko već ponudio određeni odgovor. Ono što je najbolje od svega je činjenica da drugi programeri mogu da glasaju za najbolji odgovor, tako da možete da znate koji od odgovora su zaista vredni.
- **Google Android Training** (<http://developer.android.com/training/index.html>) – „Google“ je startovao Android Training stranicu u kojoj su klase grupisane prema temama. U vreme pisanja ove knjige klasu su, uglavnom, sadržavale korisne delove koda koji se često primenjuju, što pogoduje svim Android programerima koji su prevazišli osnove programiranja. Nakon što naučite osnove koje su prikazane u ovoj knjizi, preporučujem da pogledate klase koje stoje na raspolaganju.
- **Android Discuss** (<http://groups.google.com/group/android-discuss>) – Android Discuss je diskusiona grupa koju hostuje „Google“ pomoću Google Groups servisa i u kojoj ćete moći da diskutujete o različitim aspektima Android programiranja. Ovu grupu stalno prati Android tim u „Google“ kompaniji, tako da je to dobro mesto za razjašnjavanje nekih nedoumica, dobijanje novih saveta i učenje novih trikova.

DOBIJANJE NEOPHODNIH ALATA

Pošto sada znate šta je Android i koji set funkcija pruža, verovatno želite da što pre pristupite pisanju aplikacija. Međutim, pre nego što napišete svoju prvu aplikaciju, neophodno je da preuzmete određene alate i SDK.

Za razvoj Android aplikacija možete koristiti Mac, Windows PC ili Linux računar. Svi alati koji su neophodni su potpuno besplatni i mogu se preuzeti sa Web-a. Većina primera koji su prikazani u ovoj knjizi odlično funkcioniše i u Android emulatoru, sa izuzetkom nekoliko primera koji zahtevaju pristup hardveru. Za potrebe pisanja ove knjige koristio sam Windows 7 računar da bih demonstrirao sve primere koda. Ukoliko koristite Mac ili Linux računar, slike ekrana treba da izgledaju identično; postoje određene minorne razlike, ali ćete moći da pratite tekst bez ikakvih problema.

Zabava može da počne!

JAVA JDK

Android SDK koristi Java SE Development Kit (JDK). Ukoliko na vašem računaru nije instaliran JDK, treba da ga preuzmete sa stranice koja se nalazi na adresi www.oracle.com/technetwork/java/javase/downloads/index.html i da ga instalirate pre nego što pređete na naredni odeljak.

Android SDK

Prvi i najvažniji softver koji je neophodno da preuzmete je, naravno, Android SDK. On sadrži debager, biblioteke, emulator, dokumentaciju, primere koda i uputstva.

Android SDK možete preuzeti sa stranice koja se nalazi na adresi <http://developer.android.com/sdk/index.html> (videti sliku 1-8).

The screenshot shows the 'Download the Android SDK' page. The main content area contains the following table:

Platform	Package	Size	MD5 Checksum
Windows	android-sdk_r14-windows.zip	33846273 bytes	48d44ae4cfcadade68621acb53cae80
	installer_r14-windows.exe (Recommended)	33853391 bytes	4f1cb329a41328c2cee2908b31ae6f6b
Mac OS X (intel)	android-sdk_r14-macosx.zip	30428734 bytes	812887018435382de8486f3bb26a5db4
Linux (i386)	android-sdk_r14-linux.tgz	26075938 bytes	35c989ff67184766dc4960813ede8ab5

Below the table, the page lists the following steps to set up the Android SDK:

1. Prepare your development computer and ensure it meets the system requirements.
2. Install the SDK starter package from the table above. (If you're on Windows, download the installer for help with the initial setup.)
3. Install the ADT Plugin for Eclipse (if you'll be developing in Eclipse).

SLIKA 1-8

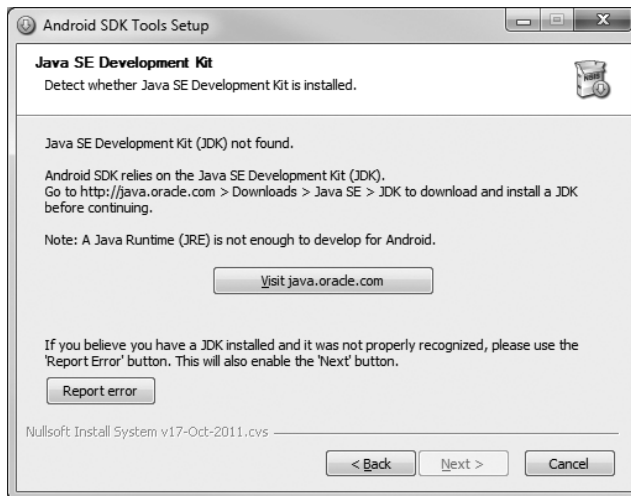
Android SDK je upakovan u zip datoteku. Možete preuzeti ovu arhivu i otpakovati njen sadržaj (android-sdk-windows folder) u folder, recimo C:\Android 4.0\. Za Windows kori-

snike „Google“ preporučuje da preuzmu `installer_r15-windows.exe` datoteku i iskoriste je za automatsko podešavanje alata. Sledeći koraci vas vode kroz ovaj postupak instaliranja.

Instaliranje Android SDK alata

Nakon što preuzmete `installer_r15-windows.exe` datoteku, dva puta je kliknite da biste inicirali proces instaliranja Android alata. Na ekranu dobrodošlice Setup čarobnjaka kliknite Next da biste nastavili dalje.

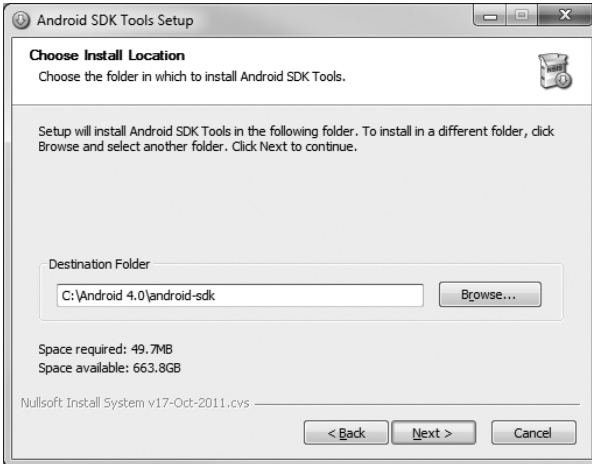
Ukoliko na vašem računaru nije instalirana Java, biće prikazan okvir za dijalog o pojavi greške, kao na slici 1-9. Međutim, čak i ukoliko je Java instalirana, može se dogoditi da bude prikazan ovaj okvir za dijalog o pojavi greške. Ukoliko se to dogodi, kliknite Report error taster, a zatim kliknite Next.



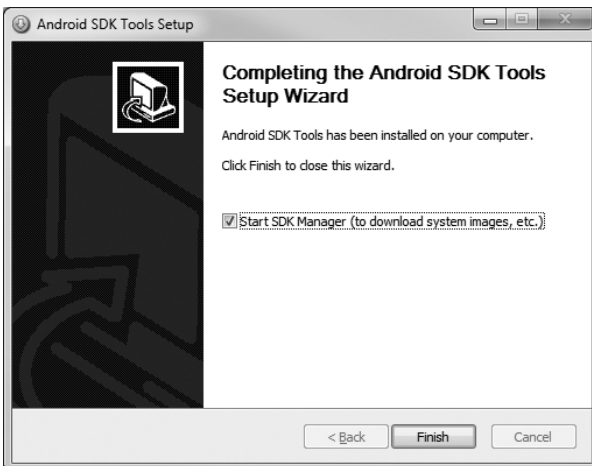
SLIKA 1-9

Od vas se zahteva da navedete odredišni folder za instaliranje Android SDK alata. Unesite određi putanju (videti sliku 1-10) i kliknite Next.

Kada bude zatraženo da izaberete Start Menu folder za kreiranje prečice programa, prihvatite podrazumevanu Android SDK Tools vrednost, a zatim kliknite Install. Nakon što se završi proces instaliranja, selektujte Start SDK Manager (to download system images, etc.) opciju i kliknite Finish (videti sliku 1-11). Tada se startuje SDK Manager.



SLIKA 1-10

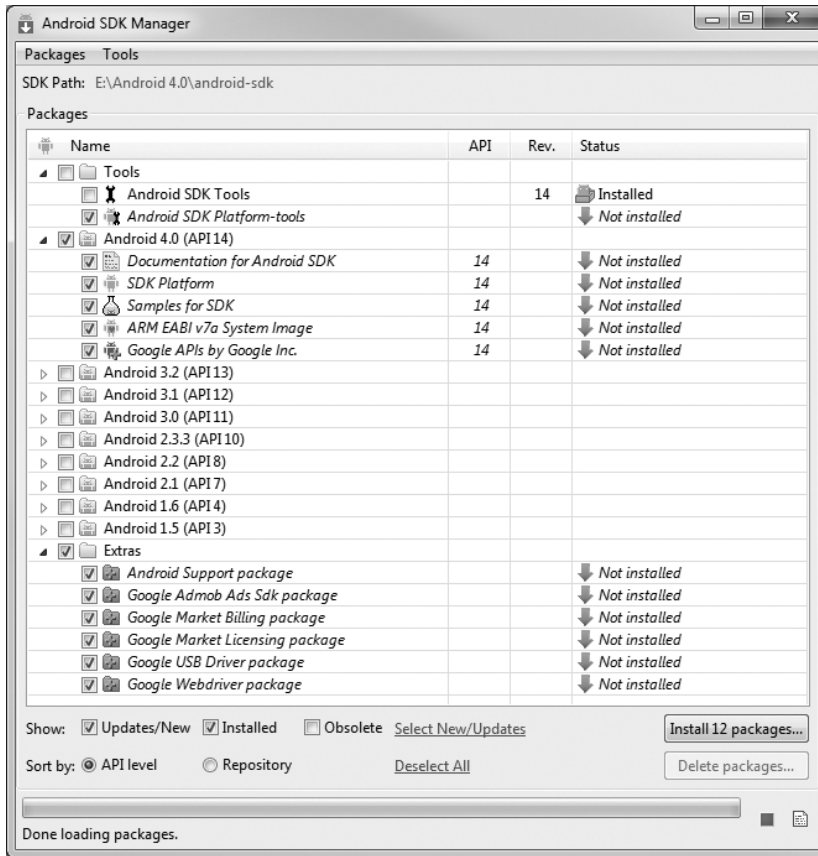


SLIKA 1-11

Konfigurisanje Android SDK menadžera

Android SDK Manager upravlja različitim verzijama Android SDK-a koje su trenutno instalirane na vašem računaru. Nakon što ga startujete, prikazuje se lista stavki; možete videti da li su trenutno instalirane na vašem računaru (videti sliku 1-12).

Selektujte relevantne alate, dokumentaciju i platforme koje želite da koristite u svom projektu. Nakon što ste selektovali sve stavke koje želite, kliknite Install taster da biste ih preuzeli. Pošto je neophodno određeno vreme da se sve što ste selektovali preuzme na lokalni računar sa Google servera, dobra ideja je da preuzmete samo ono što vam je zaista neophodno da biste započeli svoj rad, a ostatak preuzmite kada budete imali dovoljno vremena za to. Za sada, možda ćete poželeti da selektujete one stavke koje su prikazane na slici.



SLIKA 1-12



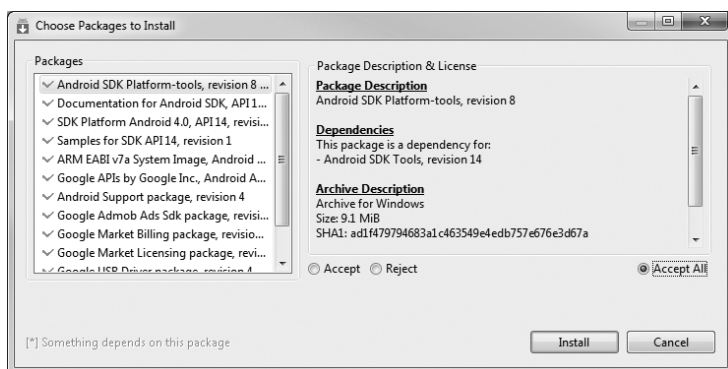
NAPOMENA Za početak treba da selektujete bar najnoviju Android 4.0 SDK platformu i dodatke. U vreme pisanja ove knjige najnovija SDK platforma je SDK Platform Android 4.0, API 14.

Svaka verzija Android operativnog sistema identifikovana je API brojem nivoa. Na primer, Android 2.3.3 je nivo 10 (API 10), dok Android 3.0 predstavlja nivo 11 (API 11) i tako redom. Za svaki nivo postoje dve platforme. Na primer, nivo 14 pruža sledeće:

- SDK platformu
- Google API interfejs kompanije „Google“

Ključna razlika između ova dve platforme ogleda se u činjenici da Google API platforma sadrži dodatne API interfejs koje je obezbedio „Google“ (kao što je Google Maps biblioteka). Zbog toga, ukoliko aplikacija koju želite da kreirate zahteva Google Maps, neophodno je da kreirate AVD korišćenjem Google API platforme (više informacija o tome možete pronaći u Poglavlju 9, „Servisi zasnovani na lokacijama“).

Od vas će se zahtevati da izaberete pakete koje želite da instalirate (videti sliku 1-13). Selektujte Accept All opciju i kliknite Install.



SLIKA 1-13

SDK Manager će nastaviti da preuzima pakete koje ste prethodno seletkovali. Instalacija može da potraje dosta dugo, tako da treba da budete strpljivi. Nakon što budu instalirani svi paketi, od vas će se zahtevati da restartujete ADB (Android Debug Bridge). Kliknite Yes.

Eclipse

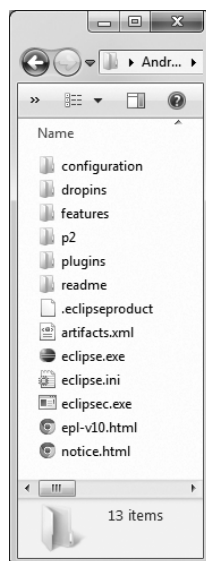
Sledeći korak je instaliranje integrisanog razvojnog okruženja (*eng.* integrated development environment, IDE). Kada je reč o Android aplikacijama, preporučuje se korišćenje Eclipse integrisanog razvojnog okruženja, koje se može koristiti za razvoj programa u različitim programskim jezicima i koje predstavlja sistem proširiv različitim dodacima. Može se koristiti za razvoj različitih tipova aplikacija korišćenjem programskih jezika, kao što su Java, Ada, C, C++, COBOL, Python i drugi.

Za potrebe razvoja Android aplikacija treba da preuzmete Eclipse IDE for Java EE Developers (www.eclipse.org/downloads/). Na raspolaganju je šest različitih izdanja:

Windows (32- i 64-bit), Mac OS X (Cocoa 32- i 64), odnosno Linux (32- and 64-bit). Jednostavno, selektujte onaj koji odgovara operativnom sistemu koji trenutno koristite. Svi primeri u ovoj knjizi su testirani korišćenjem 32-bitne verzije Eclipse integrisanog razvojnog okruženja za Windows operativni sistem.

Nakon što preuzmete Eclipse IDE, otpakujte sadržaj arhive (eclipse folder) u odgovarajući folder - na primer, C:\Android 4.0\ . Na slici 1-14 prikazan je sadržaj Eclipse foldera.

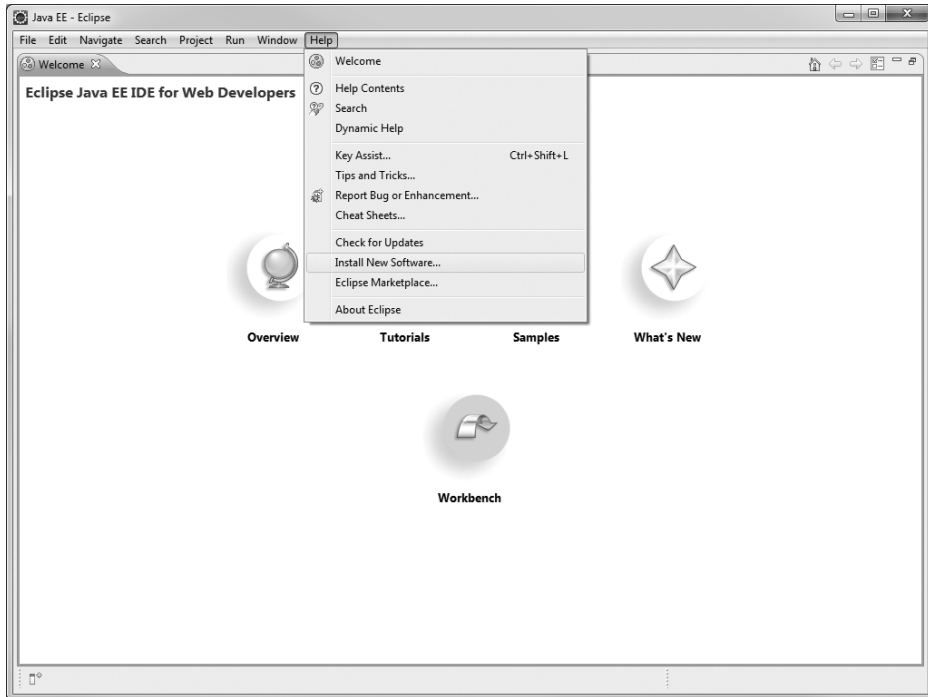
Da biste startovali Eclipse, neophodno je da dva puta kliknete eclipse.exe datoteku. Od vas se prvo zahteva da navedete radni prostor (workspace) koje ćete koristiti. U Eclipse integrisanom razvojnem okruženju radni prostor je folder u kome se čuvaju svi vaši projekti. Ne menjajte predloženu vrednost (ili možete samostalno da definišete sopstveni folder za čuvanje projekata) i kliknite OK.



SLIKA 1-14

Android Development Tools (ADT)

Nakon što startujete Eclipse integrisano razvojno okruženje, selektujte Help ⇒ Install New Software (videti sliku 1-15) da biste instalirali Android Development Tools (ADT) dodatak za Eclipse.

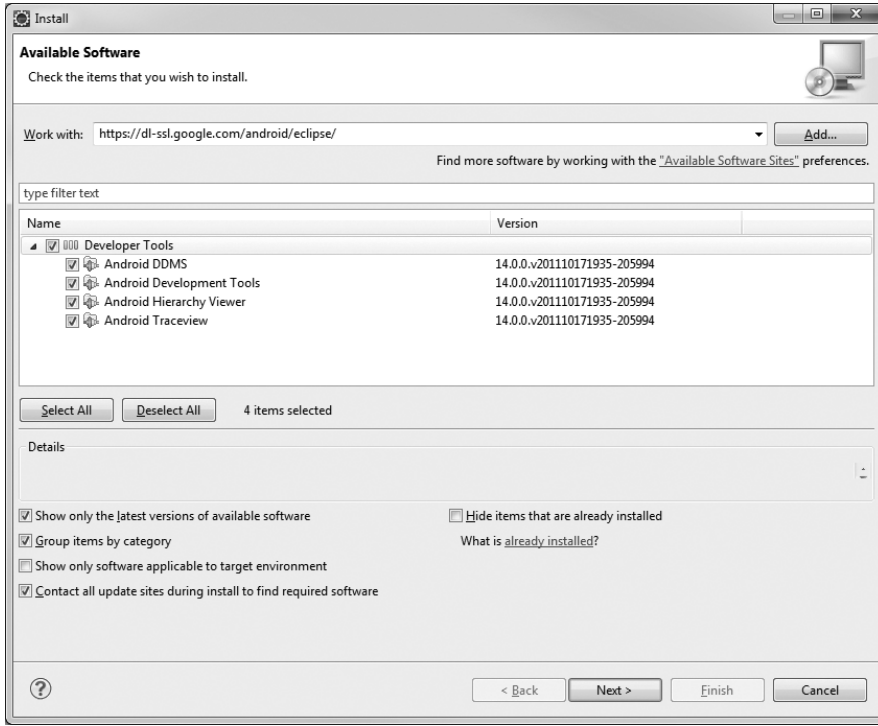


SLIKA 1-15

ADT je proširenje Eclipse integrisanog razvojnog okruženja koje podržava kreiranje i identifikovanje grešaka u Android aplikacijama. Koristeći ADT, moći ćete da u Eclipse integrisanom razvojnom okruženju uradite sledeće:

- kreirate nove projekte Android aplikacija
- pristupate alatima koji omogućavaju korišćenje Android emulatora i uređaja
- prevodite i identifikujete greške u Android aplikacijama
- eksportujete Android aplikacije u Android pakete (Android Package, skraćeno APK)
- kreirate digitalne sertifikate za potpisivanje koda vaših APK paketa

U Install okviru za dijalog koji se prikazuje navedite <https://dl-ssl.google.com/android/eclipse/> i pritisnite Enter taster. Nakon određenog vremena, biće prikazana Developer Tools stavka u središtu prozora (videti sliku 1-16). Proširite je da biste prikazali njen sadržaj: Android DDMS, Android Development Tools, Android Hierarchy Viewer i Android Traceview. Selektujte neku od raspoloživih opcija, a zatim dva puta kliknite Next.



SLIKA 1-16



NAPOMENA Ukoliko imate bilo kakvih problema prilikom preuzimanja ADT alata, pomoć možete potražiti na adresi „Google“ kompanije <http://developer.android.com/sdk/eclipse-adt.html#installing>.

Od vas se zahteva da pročitate i prihvatite licencni ugovor. Selektujte I accept the terms of the license agreements opciju, a zatim kliknite Finish. Nakon što se završi instalacija, neophodno je da restartujete Eclipse - učinite to, a zatim nastavite rad u Eclipse integrisanom razvojnom okruženju.

Nakon što se Eclipse integrisano razvojno okruženje restartuje, biće neophodno da konfigurirate Android SDK (videti sliku 1-17). Pošto ste već prethodno preuzeli Android SDK, selektujte Use existing SDKs opciju i navedite direktorijum u koji ste instalirali Android SDK. Kliknite Next.

Kada završite ovaj korak, od vas će se tražiti da pošaljete svoju statistiku korišćenja „Google“ kompaniji. Nakon što odlučite da li ćete to da uradite ili ne, kliknite Finish.



SLIKA 1-17



NAPOMENA Sa svakom novom verzijom SDK-a koja se objavi instalacioni koraci se neznatno menjaju. Ukoliko ono sa čim se susrećete prilikom instaliranja nije identično prethodno opisanom postupku, ne brinite – samo sledite instrukcije koje se prikazuju na ekranu.

Kreiranje Android virtuelnih uređaja

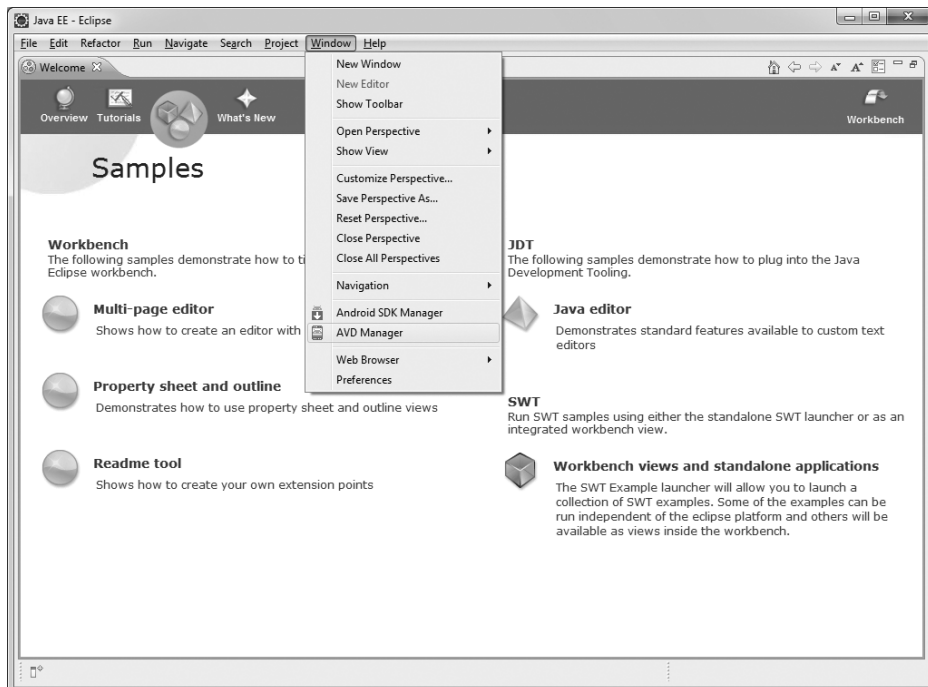
Sledeći korak je kreiranje Android virtuelnih uređaja (*eng.* Android Virtual Device, AVD) koje ćete koristiti prilikom testiranja svojih Android aplikacija. AVD je instanca emulatora koja omogućava da modelirate realni uređaj. Svaki AVD uređaj sastoji se od hardverskog profila, procedure za mapiranje u sistemsku sliku i emuliranog skladišta, kao što je secure digital (SD) kartica.

Možete da kreirate neograničeni broj AVD uređaja da biste testirali svoje aplikacije na različitim konfiguracijama. Ovo testiranje je veoma značajno zbog provere načina funkcionisanja aplikacije kada se ona izvršava na različitim uređajima koji poseduju različite funkcije.



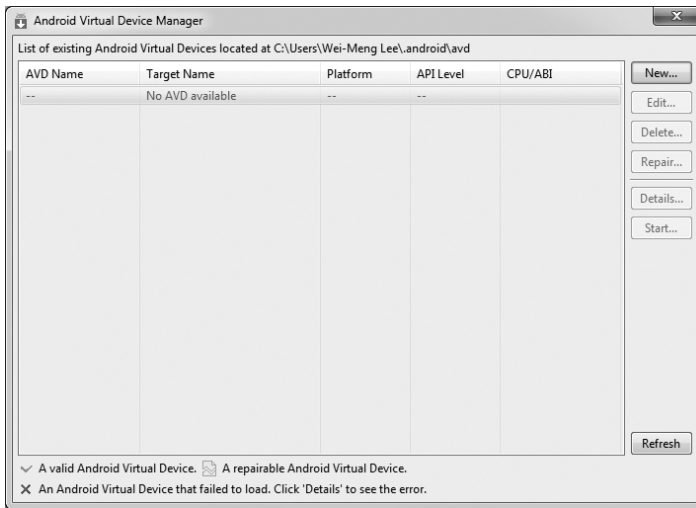
NAPOMENA Dodatak B sadrži opis određenih mogućnosti Android emulatora.

Da biste kreirali AVD uređaj, neophodno je da selektujete Window ⇒ AVD Manager stavku (videti sliku 1-18).



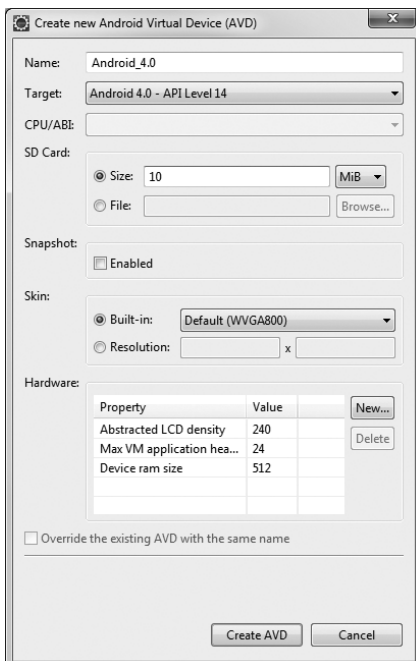
SLIKA 1-18

U Android Virtual Device Manager okviru za dijalog (videti sliku 1-19) kliknite New... taster da biste kreirali novi AVD.



SLIKA 1-19

U Create new Android Virtual Device (AVD) okviru za dijalog izvršite podešavanja onako kako je prikazano na slici 1-20. Kliknite Create AVD taster nakon što ste uneli sve neophodne podatke.



SLIKA 1-20

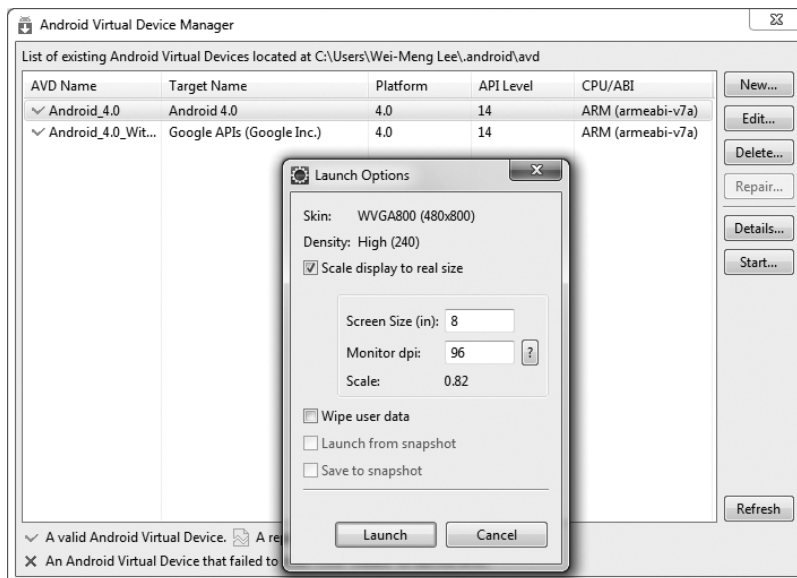
U ovoj situaciji ste kreirali jedan AVD (recimo jednostavnije, Android emulator) koji emulira Android uređaj na kome funkcioniše Android 4.0 verzija operativnog sistema i koji ima ugrađenu SD karticu kapaciteta 10 MB. Imate opciju i da emulirate uređaje sa različitim gustinama piksela na ekranu i različitim rezolucijama ekrana.



NAPOMENA Dodatak B sadrži opis načina emulacije različitih tipova Android uređaja.

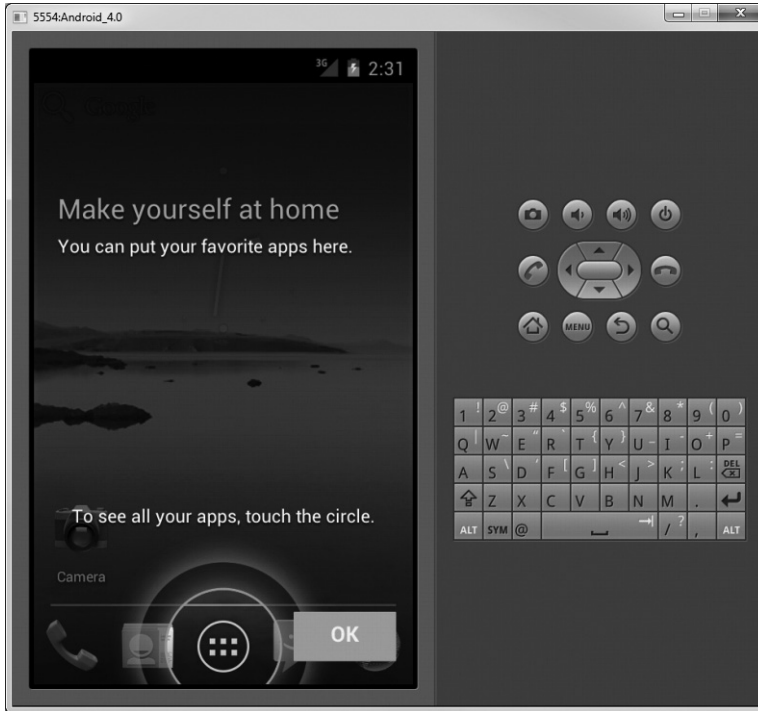
Preporučuje se da kreirate nekoliko AVD uređaja sa različitim API nivoima i hardverskim konfiguracijama, tako da vaša aplikacija može da bude testirana na različitim verzijama Android operativnog sistema.

Nakon što kreirate AVD uređaj, pravi je trenutak da ga testirate. Selektujte AVD koji želite da testirate, a zatim kliknite Start... taster. Otvara se Launch Options okvir za dijalog (videti sliku 1-21). Ukoliko imate mali monitor, preporučljivo je da selektujete Scale display to real size opciju, tako da možete da možete da prilagodite emulator manjoj veličini. Kliknite Launch taster da biste startovali emulator.



SLIKA 1-21

Startovaće se Android emulator, a, nakon određenog vremena, biće spreman za korišćenje (videti sliku 1-22). Isprobajte ga. On će funkcionisati isto kao i realni Android uređaj. Već u narednom odeljku ćete naučiti kako da napišete svoju prvu Android aplikaciju.



SLIKA 1-22

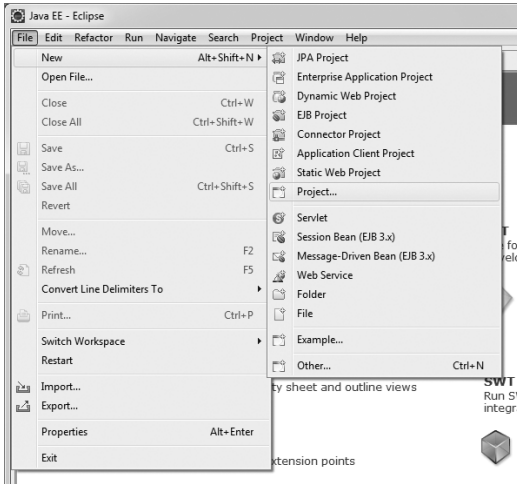
KREIRAJTE SVOJU PRVU ANDROID APLIKACIJU

Nakon što ste preuzeli i instalirali sve neophodne alate i SDK, pravi je trenutak da startujete okruženje. Kao i u svim drugim programerskim knjigama, prvi primer je čuvena Hello World aplikacija. Njen razvoj će omogućiti da detaljno sagledate sve različite komponente koje sačinjavaju svaki Android projekat.

PROBAJTE SAMI Kreiranje prve Android aplikacije

Datoteka sa kodom HelloWorld.zip može se preuzeti sa adrese Wrox.com.

1. Koristeći Eclipse integrisano razvojno okruženje, kreirajte novi projekat selektovanjem File ⇒ New ⇒ Project . . . stavke menija (videti sliku 1-23).

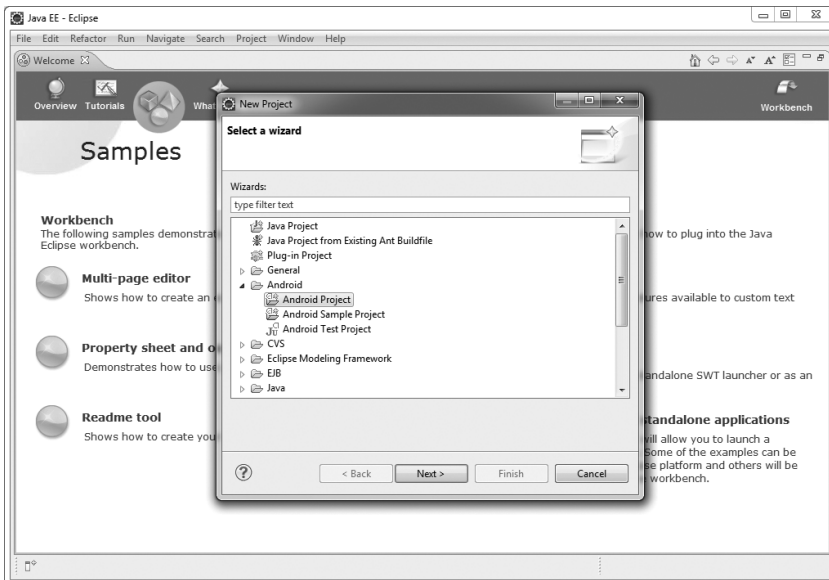


SLIKA 1-23

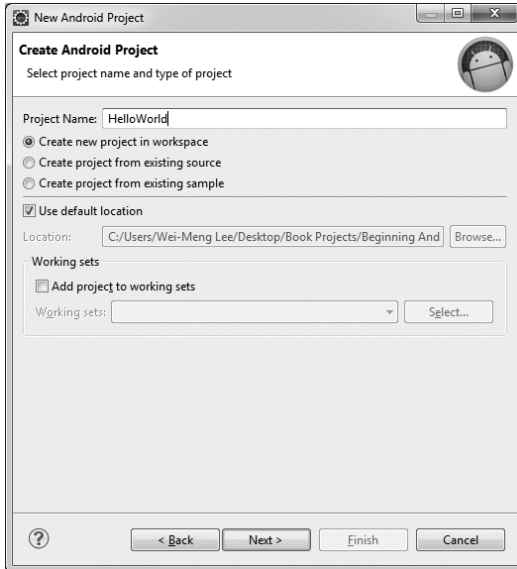


NAPOMENA Kada kreirate svoju prvu Android aplikaciju, svaki sledeći Android projekat ćete kreirati selektovanjem File ⇒ New ⇒ Android Project stavke.

2. Pristupite Android folderu i selektujte Android Project (videti sliku 1-24). Kliknite Next.
3. Nazovite svoj Android projekat HelloWorld, kao što je prikazano na slici 1-25, a zatim kliknite Next.

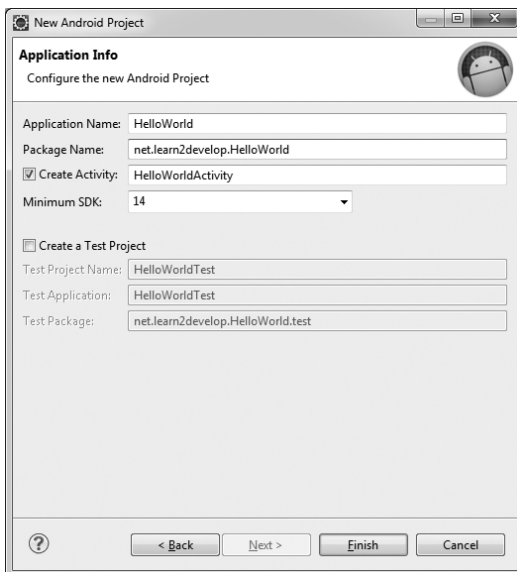


SLIKA 1-24



SLIKA 1-25

4. Selektujte Android 4.0 i kliknite Next.
5. U Application Info delu unesite detalje koji su prikazani na slici 1-26, pa kliknite Finish.

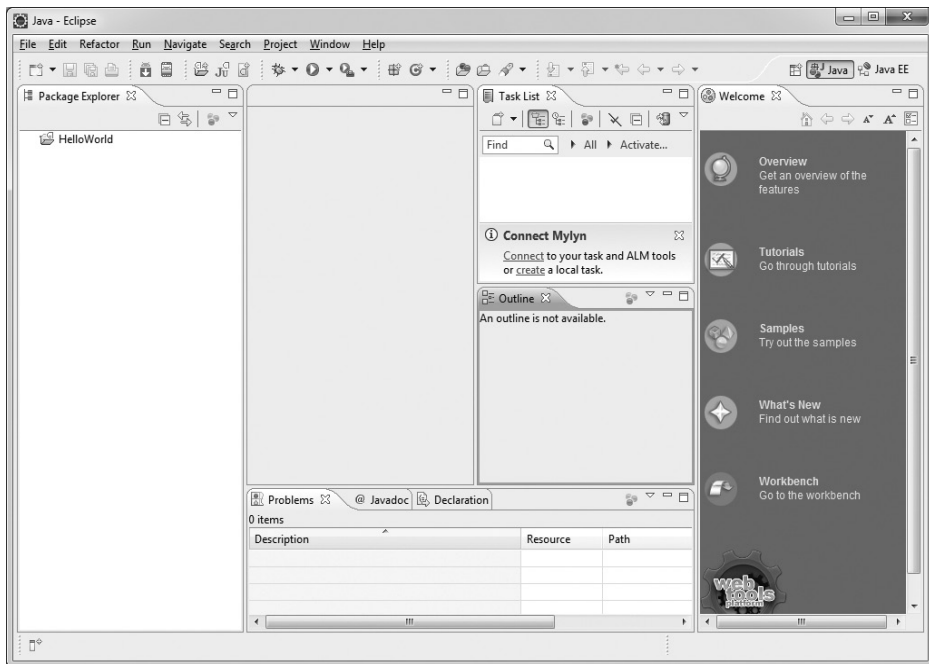


SLIKA 1-26



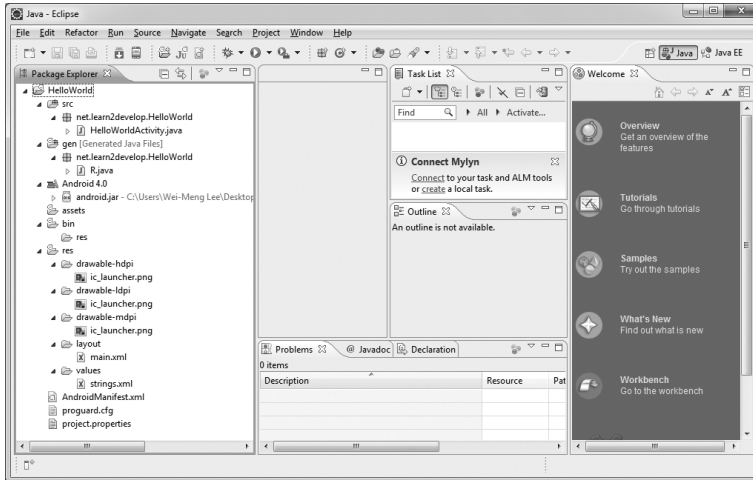
NAPOMENA Neophodno je da imate bar jednu tačku (.) u nazivu paketa. Preporučena konvencija za definisanje naziva paketa podrazumeva naziv vašeg domena u obrnutom redosledu, nakon koga se navodi naziv projekta. Na primer, domen moje kompanije je learn2develop.net; zbog toga bi naziv paketa trebalo da bude net.learn2develop.HelloWorld.

6. Eclipse integrisano razvojno okruženje treba da izgleda onako kako je prikazano na slici 1-27.

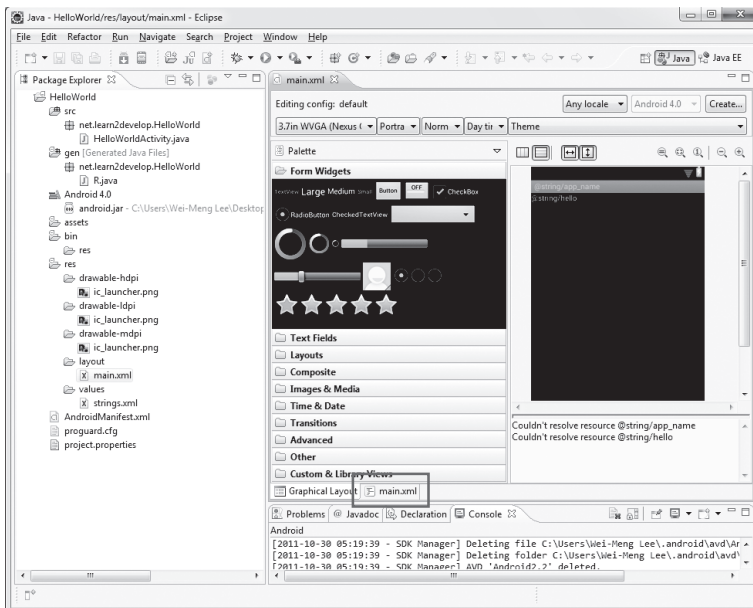


SLIKA 1-27

7. U Package Explorer panelu (lociranom u levom delu Eclipse integrisanog razvojnog okruženja) pristupite HelloWorld projektu, tako što ćete kliknuti različite strelice koje su prikazane sa leve strane svake stavke u projektu (videti sliku 1-28). U `res/layout` folderu dva puta kliknite `main.xml` datoteku.
8. Datoteka `main.xml` definiše korisnički interfejs (*eng.* user interface, UI) vaše aplikacije. Podrazumevani način prikaza je Layout, koji grafički prikazuje aktivnost. Da biste ručno modifikovali korisnički interfejs, kliknite `main.xml` karticu, koja je locirana u donjem delu (videti sliku 1-29).



SLIKA 1-28



SLIKA 1-29

9. Dodajte u main.xml datoteku sledeći kod, prikazan podebljanim slovima:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/
android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
```



```
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello" />

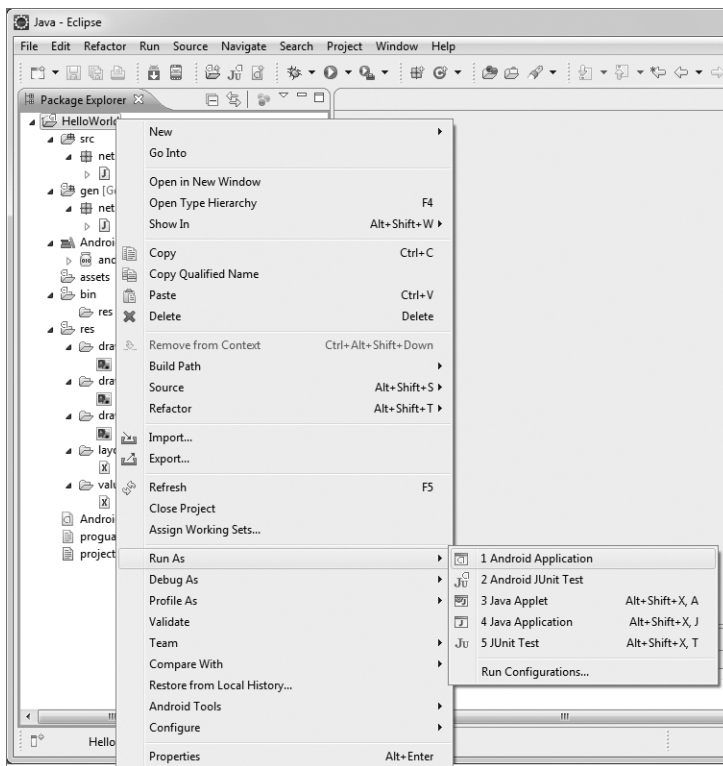
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="This is my first Android Application!" />

<Button
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="And this is a clickable button!" />

</LinearLayout>
```

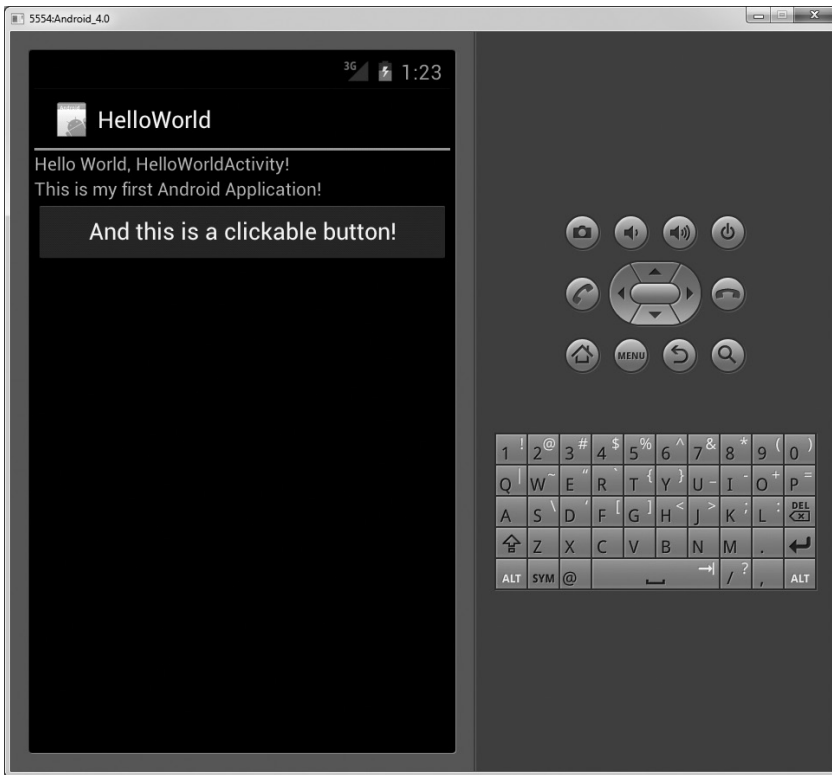
10. Da biste snimili izvršene promene u projektu, pritisnite kombinaciju tastera Ctrl+S.

11. Sada možete da pristupite testiranju svoje aplikacije u Android emulatoru. Desnim tasterom miša kliknite naziv projekta u Eclipse integrisanom razvojnom okruženju, a zatim selektujte Run As ⇒ Android Application stavku (videti sliku 1-30).



SLIKA 1-30

12. Ukoliko niste napravili ni jednu grešku u projektu, aplikacija će se instalirati i startovati u Android emulatoru (videti sliku 1-31).



SLIKA 1-31

13. Kliknite Home taster (ikonice u obliku kuće, koja se nalazi u donjem levom uglu iznad tastature), tako da se prikaže Home ekran (videti sliku 1-32).
14. Kliknite ikonicu za startovanje aplikacije da biste prikazali listu aplikacija koje su instalirane na vašem uređaju. Obratite pažnju da je sada HelloWorld aplikacija prikazana u listi aplikacija na ekranu koji omogućava njihovo startovanje (application launcher); videti sliku 1-33.



SLIKA 1-32

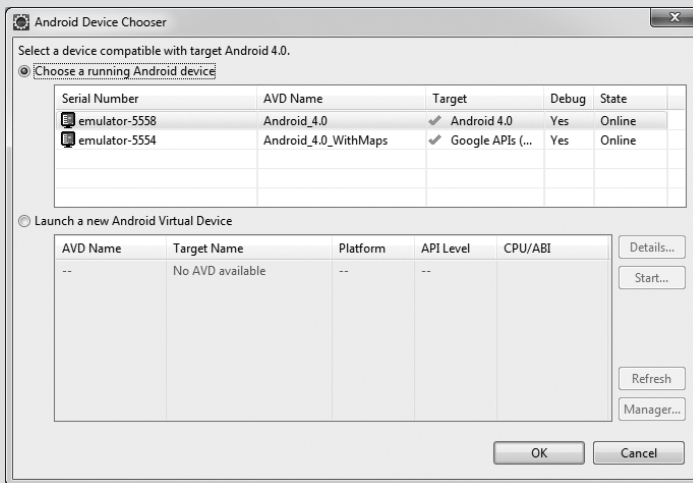


SLIKA 1-33

KOJI AVD će se koristiti prilikom testiranja aplikacije?

Recimo da ste prethodno kreirali nekoliko AVD uređaja korišćenjem AVD menadžera. Koji od njih će koristiti Eclipse integrisano razvojno okruženje prilikom izvršavanja vaše Android aplikacije? Eclipse proverava određite koje ste specifikovali (prilikom kreiranja novog projekta), poredeći ga sa listom AVD uređaja koje ste prethodno kreirali. Prvi od njih koji odgovara će biti korišćen prilikom izvršavanja vaše aplikacije.

Ukoliko imate više odgovarajućih AVD uređaja koje ste prethodno koristili za testiranje aplikacije, u Eclipse integrisanom razvojnom okruženju će biti prikazan Android Device Chooser okvir za dijalog, koji omogućava da selektujete željeni emulator/uređaj za debugiranje vaše aplikacije (videti sliku 1-34).



SLIKA 1-34

Način funkcionisanja

Da biste kreirali Android projekat u Eclipse integrisanom razvojnom okruženju, neophodno je da navedete informacije koje su prikazane u tabeli 1-2.

TABELA 1-2 Projektne datoteke koje se kreiraju na osnovu inicijalnih podešavanja

Svoјstvo	Opis
Project name	Naziv projekta
Application name	Korisnički definisan naziv aplikacije
Package name	Naziv paketa; treba da koristite notaciju zasnovanu na inverznom prikazivanju naziva domena
Create Activity	Naziv prve aktivnosti u vašoj aplikaciji
Min SDK Version	Minimalna verzija SDK koju koristite u projektu

U Android operativnom sistemu aktivnost predstavlja prozor koji sadrži korisnički interfejs u aplikacijama. Može se dogoditi da određena aplikacija nema ni jednu ili da ima više aktivnosti. U našem primeru aplikacija sadrži aktivnost `HelloWorldActivity` – ona je polazna tačka aplikacije (prikazuje se kada se startuje aplikacija). U Poglavlju 2 naći ćete detaljan opis aktivnosti.

U ovom jednostavnom primeru neophodno je da modifikujete `main.xml` datoteku da bili prikazani string `This is my first Android Application!` i jedan taster. Datoteka `main.xml` sadrži jednu aktivnost, koja se prikazuje kada se učita `HelloWorldActivity`.

Prilikom debagovanja aplikacije u Android emulatoru, aplikacija se automatski instalira u emulatoru. I to je sve – napravili ste svoju prvu Android aplikaciju!

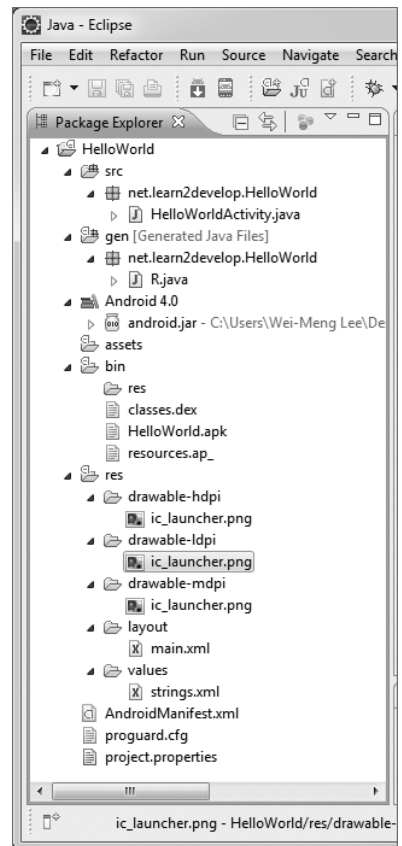
U sledećem odeljku opisano je kako različite datoteke u Android projektu funkcionišu zajedno da bi se omogućilo da se aplikacija ispravno izvršava.

ANATOMIJA ANDROID APLIKACIJE

Pošto ste kreirali svoju prvu Hello World aplikaciju u Android operativnom sistemu, pravi je trenutak da razmotrimo koji su elementi ovog Android projekta i kako pojedini od njih omogućavaju funkcionisanje aplikacije. Prvo, primetićete da su različite datoteke koje sačinjavaju Android projekat prikazane u Package Explorer panelu u Eclipse integrisanom razvojnom okruženju (videti sliku 1-35).

Postoje sledeći folderi i u njima odgovarajuće datoteke:

- `src` – Sadrži `.java` izvorne datoteke u projektu. U ovom primeru postoji samo jedna datoteka - `HelloWorldActivity.java`. Ona je izvorna datoteka za vašu aktivnost. U njoj pišete kod za funkcionisanje vaše aplikacije. Java datoteka je prikazana u okviru naziva paketa projekta, koji je u ovom primeru `net.learn2develop.HelloWorld`.
- `gen` – Sadrži `R.java` datoteku, koju je generisao prevodilac, a koja referencira sve resurse u projektu. Nju ne treba da modifikujete. Svi resursi u projektu se automatski prevode u ovu klasu, tako da možete da ih referencirate njenim korišćenjem.
- `Android 4.0 library` – Ova stavka sadrži samo jednu datoteku - `android.jar`, koja sadrži sve biblioteke klasa neophodne za jednu Android aplikaciju.
- `assets` – Ovaj folder sadrži sva sredstva koja koristi vaša aplikacija, kao što su HTML, tekstualne datoteke, baze podataka itd.
- `bin` – Ovaj folder sadrži datoteke koje je kreirao ADT u procesu prevođenja. Konkretno, generiše se `.apk` datoteka (Android Package datoteka). Datoteka sa ekstenzijom `.apk` je binarni kod Android aplikacije. Ona sadrži sve što je neophodno za izvršavanje jedne Android aplikacije.



SLIKA 1-35

- `res` – Ovaj folder sadrži sve resurse koji se koriste u aplikaciji. Sadrži i druge potfoldere: `drawable-<resolution>`, `layout` i `values`. U Poglavlju 3 detaljno je opisan postupak definisanja različitih rezolucija i gustina za uređaje.
- `AndroidManifest.xml` – Ovo je manifest datoteka za Android aplikaciju. U njoj definišete privilegije koje su neophodne za vašu aplikaciju, kao i ostale funkcije (kao što su filter sadržaja, primaoci i slično). U Poglavlju 2 detaljno je opisan postupak korišćenja `AndroidManifest.xml` datoteke.

Datoteka `main.xml` definiše korisnički interfejs za aktivnost. Posebno razmotrite tekst prikazan podebljanim slovima:

```
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello" />
```

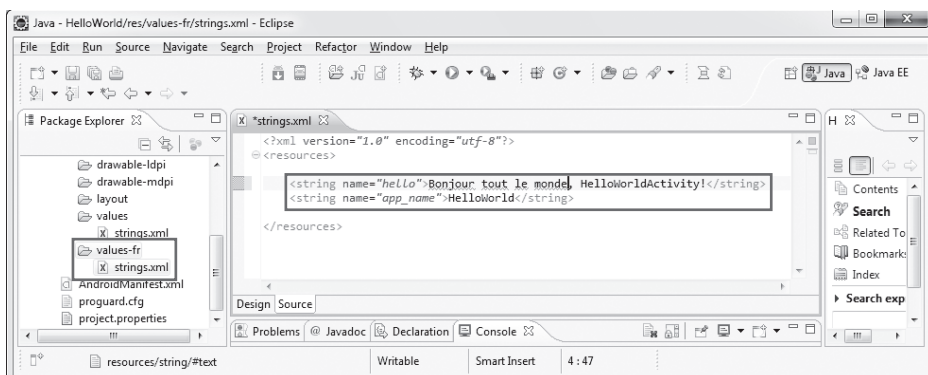
U ovom primeru `@string` ukazuje na `strings.xml` datoteku koja je locirana u `res/values` folderu. Zbog toga, `@string/hello` ukazuje na `hello` string koji je definisan u `strings.xml` datoteci, a to je, zapravo, Hello World, `HelloWorldActivity!`

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="hello">Hello World, HelloWorldActivity!</string>
    <string name="app_name">HelloWorld</string>

</resources>
```

Preporučuje se da sve stringove koje koristite u svojoj aplikaciji snimate u `strings.xml` datoteci i referencirate ih korišćenjem `@string` identifikatora. Tako će, ukoliko nekada budete želeli da svoju aplikaciju lokalizujete na neki drugi jezik, biti potrebno samo da iskopirate celokupan `values` folder i modifikujete vrednosti `strings.xml` datoteke da bi ona sadržavala string na jeziku koji želite da prikazujete na ekranu uređaja. Na slici 1-36 prikazan je dodatni folder pod nazivom `values-fr`, koji sadrži `strings.xml` datoteku sa istim `hello` stringom na francuskom jeziku.



SLIKA 1-36

Ukoliko korisnik učita istu aplikaciju na telefonu koji je konfigurisan tako da prikazuje poruke na francuskom jeziku na osnovu inicijalnih podešavanja, vaša aplikacija će automatski da prikazuje `hello string` na francuskom jeziku.

Sledeća značajna datoteka u Android projektu je manifest datoteka. Obratite pažnju na sadržaj `AndroidManifest.xml` datoteke:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="net.learn2develop.HelloWorld"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk android:minSdkVersion="14" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <activity
            android:label="@string/app_name"
            android:name=".HelloWorldActivity" >
            <intent-filter >
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
            />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

`AndroidManifest.xml` datoteka sadrži detaljne informacije o aplikaciji:

- Definiše naziv paketa aplikacije u obliku `net.learn2develop.HelloWorld`.
- Kod verzije aplikacije je 1 (definiše se pomoću `android:versionCode` atributa). Ova vrednost se koristi za identifikovanje broja verzije aplikacije. Ona se može koristiti za programsko utvrđivanje da li je neophodno ažuriranje aplikacije.
- Naziv verzije aplikacije je 1.0 (definiše se pomoću `android:versionName` atributa). Vrednost ovog stringa se, obično, koristi za prikazivanje krajnjem korisniku. Treba da koristite `<major>.<minor>.<point>` format.
- Atribut `android:minSdkVersion` elementa `<uses-sdk>` specificira minimalnu verziju operativnog sistema na kojoj će se izvršavati aplikacija.
- Aplikacija koristi sliku `ic_launcher.png`, koja je locirana u `drawable` folderu.
- Naziv ove aplikacije je određen stringom pod nazivom `app_name`, koji je definisan u `strings.xml` datoteci.
- Postoji jedna aktivnost u ovoj aplikaciji koja je predstavljena `HelloWorldActivity.java` datotekom. Labela prikazana za ovu aktivnost ima isti naziv kao i aplikacija.

- U okviru definicije za ovu aktivnost postoji element pod nazivom `<intent-filter>`:
 - Kategorija za filter sadržaja je definisana kao `android.intent.action.MAIN` da bi se ukazalo da ova aktivnost služi kao početna tačka aplikacije.
 - Kategorija za filter sadržaja je definisana kao `android.intent.category.LAUNCHER` da bi se ukazalo da aplikacija može da se startuje pomoću launcher ikone uređaja. Ovo je detaljno opisano u Poglavlju 2.
 - Kada budete dodavali još datoteka i foldera u svoj projekat, Eclipse integrisano razvojno okruženje će automatski generisati dodatni sadržaj `R.java` datoteke, koja trenutno sadrži sledeće:

```

/* AUTO-GENERATED FILE. DO NOT MODIFY.
 *
 * This class was automatically generated by the
 * aapt tool from the resource data it found. It
 * should not be modified by hand.
 */

package net.learn2develop.HelloWorld;

public final class R {
    public static final class attr {
    }
    public static final class drawable {
        public static final int ic_launcher=0x7f020000;
    }
    public static final class layout {
        public static final int main=0x7f030000;
    }
    public static final class string {
        public static final int app_name=0x7f040001;
        public static final int hello=0x7f040000;
    }
}

```

Nije predviđeno da samostalno menajete sadržaj `R.java` datoteke; Eclipse okruženje će automatski generisati njen sadržaj uvek kada na bilo koji način modifikujete svoj projekat.



NAPOMENA Ukoliko ručno obrišete `R.java` datoteku, Eclipse integrisano razvojno okruženje će je odmah ponovo generisati za vas. Obratite pažnju da Eclipse okruženje može da generiše `R.java` datoteku samo ukoliko projekat ne sadrži ni jednu grešku. Ukoliko Eclipse okruženje nije ponovo generisalo `R.java` datoteku nakon što ste je obrisali, neophodno je da detaljno proverite svoj projekat. Možda kod sadrži određene sintaksne greške ili možda vaše XML datoteke (kao što je `AndroidManifest.xml`, `main.xml` itd) nisu dobro formirane.

Konačno, kod koji povezuje aktivnost sa korisničkim interfejsom (`main.xml`) je `setContentTextView()` metod, koji je definisan sledećom `HelloWorldActivity.java` datotekom:


```
package net.learn2develop.HelloWorld;

import android.app.Activity;
import android.os.Bundle;

public class HelloWorldActivity extends Activity {
    /** Izvršava se kada se aktivnost prvi put obavlja. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Ovde `R.layout.main` ukazuje na `main.xml` datoteku koja je locirana u `res/layout` folderu. Pošto dodajete XML datoteke u `res/layout` folder, njihovi nazivi će se automatski generisati u `R.java` datoteci. Metod `onCreate()` je jedan od brojnih metoda koji se aktiviraju kada se obavi aktivnost. U Poglavlju 2 detaljno je opisan „životni ciklus“ aktivnosti.

PREGLED POGLAVLJA

U ovom poglavlju dat je kratak pregled Android jezika, uz poseban naglasak na njegove mogućnosti. Ukoliko ste sledili navedena uputstva kako se preuzimaju alat i Android SDK, sada treba da pred sobom imate funkcionalan sistem – onaj koji omogućava da razvijate mnogo interesantnije Android aplikacije u odnosu na prikazanu Hello World aplikaciju. U narednom poglavlju ćete naučiti nešto više o osnovnim konceptima aktivnosti i sadržaja i o veoma značajnim ulogama koje one imaju u razvoju Android aplikacija.

VEŽBE

1. Šta je AVD?
2. Koja je razlika između `android:versionCode` i `android:versionName` atributa u `AndroidManifest.xml` datoteci?
3. Za šta se koristi `strings.xml` datoteka?

Odgovore na pitanja i rešenja vežbi možete pronaći u Dodatku C.

ŠTA STE NAUČILI U OVOM POGLAVLJU

Tema	Ključni koncepti
Android operativni sistem	Android mobilni operativni sistem sa otvorenim kodom, koji je zasnovan na Linux operativnom sistemu. Raspoloživ je svakom ko želi da ga adaptira i izvršava na sopstvenim uređajima.
Jezici koji se koriste za razvoj Android aplikacija	Koristićete Java programski jezik za razvoj Android aplikacija. Napisane aplikacije se prevode u Dalvik izvršne datoteke, koje se, nakon toga, izvršavaju pomoću Dalvik virtuelne mašine.
Android Market	Android Market sadrži sve različite Android aplikacije koje pišu nezavisni autori softvera.
Alati za razvoj Android aplikacija	Eclipse integrisano razvojno okruženje, Android SDK i ADT
Aktivnosti	Aktivnost je predstavljena ekranom u Android aplikaciji. Moguće je da svaka aplikacija nema ni jednu ili da ima više aktivnosti.
Android manifest datoteka	AndroidManifest.xml datoteka sadrži detaljne informacije o konfiguraciji aplikacije. Kako primeri budu postajali složeniji, biće neophodno da modifikujete ovu datoteku; u poglavljima koja slede videćete koje različite informacije možete da dodajete.
