



Funded by the
Erasmus+ Programme
of the European Union



This project has been funded with support from the European Commission. This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



KI205 - JAVA 8: PROGRAMIRANJE U JAVI NA ANDROID PLATFORMI

Uvod u Android programiranje

Lekcija 01

PRIRUČNIK ZA STUDENTE

KI205 - JAVA 8: PROGRAMIRANJE U JAVI NA ANDROID PLATFORMI

Lekcija 01

UVOD U ANDROID PROGRAMIRANJE

- ✓ Uvod u Android programiranje
- ✓ Poglavlje 1: Uvod u Android OS
- ✓ Poglavlje 2: Alati i tehnologije za razvoj Android aplikacija
- ✓ Poglavlje 3: Android emulator Genymotion
- ✓ Poglavlje 4: Domaći zadatak 1
- ✓ Zaključak

Copyright © 2017 – UNIVERZITET METROPOLITAN, Beograd. Sva prava zadržana. Bez prethodne pismene dozvole od strane Univerziteta METROPOLITAN zabranjena je reprodukcija, transfer, distribucija ili memorisanje nekog dela ili čitavih sadržaja ovog dokumenta., kopiranjem, snimanjem, elektronskim putem, skeniranjem ili na bilo koji drugi način.

Copyright © 2017 BELGRADE METROPOLITAN UNIVERSITY. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without the prior written permission of Belgrade Metropolitan University.

UVOD

Android operativni sistem predstavlja najrasprostranjeniju distribuciju Linux operativnog sistema.

Cilj predmeta jeste sticanje znanja i veština neophodnih za razumevanje koncepata i principa razvoja softvera za mobilne uređaje. Takođe, akcenat će biti stavljen na specifične alate i tehnologije koji se koriste u industriji softvera namenjenog mobilnim uređajima.

Kao ishod predmeta ističu se poznavanje navedenih alata i tehnologija i potpuno razumevanje koncepata i principa mobilnog računarstva.

U fokusu predmeta biće Android operativni sistem i aplikacije koje se izvršavaju upravo na njemu. Android predstavlja, danas, najrašireniju distribuciju Linux operativnog sistema. Istorijat operativnog sistema počinje od 2005. godine kada je kompanija *Android INC.* Razvila prvu verziju operativnog sistema. Kroz vizionarska razmišljanja, da će se savremeno računarstvo kretati sve više ka razvoju mobilnih uređaja i povećanju mobilnosti, kompanija *Google* otkupljuje prava na Android operativni sistem i preuzima odgovornost za njegov dalji razvoj.

Pored vlastitih razvojnih timova, kompanija *Google*, za unapređenje i razvoj operativnog sistema, brzo stiče saveznike u kompanijama koje su bile direktno pogođene ekspanzijom operativnog sistema IOS za *Apple* mobilne uređaje. U početku se

radilo o kompanijama: *Motorola* i *SonyEricsson*, a danas su glavni nosioci primene Android operativnog sistema kompanije: *Samsung*, *LG*, *Sony*, kao i kineski giganti *Huawei* i *Xiaomi*.

Veoma brzo Android stiče veoma moćnu i široko rasprostranjenu programersku zajednicu okupljenu oko developer.android.com u okviru koje se:

- veoma brzo dobijaju odgovori na sva pitanja od interesa za korisnike i programere;
- obezbeđuju Android treninzi za aktuelne i buduće programere;
- Iznose iskustva i zapažanja u vezi sa prednostima i nedostacima aktuelnih verzija operativnog sistema i sl.

✓ Poglavlje 1

Uvod u Android OS

VERZIJE ANDROID OS

Sve verzije Android operativnog sistema nose zvaničan naziv dobijen po nekoj poslastici.

Sledećom tabelom prikazan je istorijat **Google Android** operativnog sistema sa funkcionalnostima koje su nove verzije uvele (izvor: <http://www.blic.rs/IT/514074/Sve-verzije-Android-OS-Od-G1-do-Lollipopa>).

Slika-1 Verzije Android operativnog sistema

Slika 1.1.1 Verzije Android operativnog sistema

ANDROID 6.0

U 2015. predstavljen je Android 6.0 sa brojnim novinama.

U 2015. godini kompanija *Google* počela je sa distribucijom najnovije generacije operativnog sistema Android pod nazivom **Android 6 – Marshmallow**. Ovaj operativni sistem uvodi brojne novine:

- **Android NOW ON TAP:** mogućnost otvaranja dodatne opcije bilo koje aplikacije ili ekrana zadržavajući Android Home taster.
- **Dozvole za aplikacije:** Mogućnost odlučivanja o tome koja aplikacija će pristupati kojim opcijama telefona (kontakti, mikrofona, poruke, memorija, veb, kartica i sve drugo).
- **Podrška za senzore otiska prsta:** Sad gotovo svaki Android može da ima skener otiska prsta, ako to proizvođač poželi da stavi na svoj uređaj.
- **Doze Mode:** Nova opcija za uštedu potrošnje baterije.
- **Novi meni aplikacija:** Ikonice aplikacija u glavnom meniju se prikazuju sa poboljšanjima koje uključuju aplikacije koje se najčešće koriste, predlog Androida koje aplikacije da koristite i pretragu aplikacija.

U toku je ažuriranje premium uređaja ovom verzijom Android operativnog sistema.

Slika 1.1.2 Android 6.0

KARAKTERISTIKE ANDROID OS

Unificirani pristup razvoju aplikacija je jedna od glavnih prednosti Android operativnog sistema.

Osnovna prednost Android operativnog sistema jeste postojanje unificiranog pristupa razvoju aplikacija. Sva softverska rešenja se razvijaju pod Android platformom i izvršavaju na uređajima koje pokreće neka od verzija Android operativnog sistema. Ovaj operativni sistem pripada tzv. *open-source* konceptu, a to znači da su alati i tehnologije za prilagođavanje i unaprećenje operativnog sistema, kao i aplikacija koje se izvršavaju pod Androidom, dostupni potpuno besplatno. Otuda, **ne postoje posebno određene hardversko – softverske konfiguracije za Android** ali moraju biti ispoštovani sledeći zahtevi:

- skladištenje podataka jednostavnom SQLite relacionom bazom podataka;
- pristupanje mobilnim mrežama putem svih poznatih mobilnih standarda (GSM, GPRS, EDGE, UMTS, LTE) kao i umrežavanje putem Bluetooth, NFC, WiFi, WiMax i drugih poznatih standarda;
- slanje tekstualnih i multimedijalnih poruka;
- web čitač;
- hardverska podrška za akcelerometar, GPS, kameru i sl.
- podrška za ekrane osetljive na dodir;
- multi-tasking;
- deljenje Internet konekcije itd.

ARHITEKTURA ANDROID OS

Arhitektura Androida organizovana je po slojevima.

Android operativni sistem podeljen je na slojeve kao što je prikazano sledećom slikom (izvor: <http://www.plagosus.net>).

Slika 1.1.3 Slojevi Android OS

- **Linux jezgro** predstavlja osnovu operativnog sistema i u njemu su sadržani pokretački programi svih hardverskih komponenta integrisanih u mobilni uređaj;
- **Biblioteke** čuvaju programske kodove svih modula neophodnih za funkcionisanje operativnog sistema;
- **Android Runtime** je istog nivoa kao i biblioteke i obezbeđuje arhive pomoću kojih JAVA programeri pišu Android aplikacije;
- **Radni okvir aplikacija** obezbeđuje različite funkcionalnosti Android operativnog sistema bez kojih kreirane aplikacije ne bi mogle da se izvršavaju;
- **Nivo aplikacija** obuhvata sve aplikacije koje su pre-instalirane na uređaju, preuzete sa Android Marketa ili drugih izvora ili samostalno kreirane i implementirane.

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

✓ 1.1 Primer 1 - Razvoj prve Android aplikacije

KREIRANJE ANDROID VIRTUELNOG UREĐAJA

AVD je instanca emulatora koja omogućava modeliranje realnog Android uređaja.

Pod AVD-om podrazumeva se instancu emulatora, koja modeluje neki stvarni Android uređaj. Svaki AVD sadrži parametre koji ga opisuju – hardverski profil uređaja, verzija Androida koja je instalirana u emulatoru, kao i veličina SD kartice koju emulator poseduje.

Moguće je napraviti veći broj AVD-ova za testiranje aplikacija na više različitih konfiguracija. Android aplikacije se mogu testirati direktno na android telefonu (ili tabletu), ali osnovna prednost korišćenja AVD je što je moguće raditi testove da bi se proverilo kako se aplikacija ponaša na uređajima sa različitim sposobnostima i na raznim verzijama Androida.

Android virtuelni uređaj se kreira u nekoliko koraka:

- U Eclipse IDE razvojnom okruženju bira se opcija *AVD Manager* iz menija Window.

Navedeno je prikazano sledećom slikom.

Slika 1.2.1 Otvaranje AVD menadžera

DEFINISANJE KONKRETNOG AVD

Klikom na New počinje kreiranje nove AVD instance.

U okviru Menadžera neophodno je kliknuti na New da bi bila kreirana nova AVD instancu (sledeća slika).

Slika 1.2.2 Kreiranje nove AVD instance

DEFINISANJE KONKRETNOG AVD - NASTAVAK

Za svaki AVD neophodno je podesiti odgovarajuće parametre.

Podešavaju se sledeći parametri:

AVD Name: Ime po kome se raspoznaje ova instancu (telefonJB)

Device: Tip uređaja koji se emulira.

Target: Ovde se bira verziju Androida koja će biti emulirana. Moguće je izabrati samo one verzije za koje je instaliran API u okviru SDK Menadžera. Ako pored verzije stoji Google API,

znači da će u emulatoru biti dostupni i neki guglovi servisi (kao na primer Google Maps), a u suprotnom biće emuliran 'čist' Android.

Keyboard: Daje mogućnost korišćenja fizičke tastature u okviru emulatora. U suprotnom će za svako tekst polje iskakati virtuelna tastatura koja se koristi pomoću miša.

Skin: Obezbeđuje okvir sa standardnim android tasterima (Back, Home, Menu) kada se pokrene AVD.

Front Camera i Back Camera: Ako se izabere veb kamera oponaša kameru telefona.

Memory Options: Preporučljivo je da bude bar 512Mb RAM (deo ovoga će zauzeti sam Android sistem), a VM Heap će biti zadržan na vrednosti 32.

Internal Storage: Definiše vrednost za kapacitet internog skladišta memorije (za početak 200 MB će biti dovoljno).

SD Card: Definiše vrednost za kapacitet eksternog skladišta memorije.

Emulation Options: Izбором Use Host GPU emulator će raditi brže.

AVD POKRETANJE

Kreiran Android emulator koristi se za testiranje aplikacija.

Klikom na OK biće kreiran Android Emulator, i u okviru AVD Menadžera, klikom na Start pa na Launch biće pokrenut (sledeća slika).

Sada je sve spremno za kreiranje prve Android aplikacije.

Slika 1.2.3 Kreirana AVD instanca

PRVA ANDROID APLIKACIJA

Android aplikacije imaju formu projekata.

Razvoj svake Android aplikacije započinje pokretanjem Eclipse IDE razvojnog pakovanja. Nakon toga iz menija *File*, bira se opcija *New*, za zatim *Project*.

Slika 1.2.4 Novi Android projekat

Nakon klika na opciju *Project* otvara se prozor u kojem se, u meniju *Android*, bira opcija *Android Project*.

Slika 1.2.5 Izbor opcije Android projekat

KREIRANJE ANDROID PROJEKTA

Svi fajlovi čuvaju se u folderu projekta.

U sledećem koraku neophodno je dodeliti naziv projektu, aplikaciji i paketu koji će čuvati dokumentaciju projekta. Takođe, biće izabran i API koji odgovara verziji Android operativnog sistema za koju se aplikacija razvija. U konkretnom slučaju ciljani API je verzija Android 4.4 KitKat, kompajliranje će biti obavljeno najnovijom verzijom za Android 6.0 (sledeća slika). Ključne informacije o aplikaciji su sledeće:

Naziv projekta: PrvaAdroidAplikacija;

Naziv aplikacije: PrvaAdroidAplikacija;

Naziv paketa: com.Metropolitan.prvaadroidaplikacija.

Klikom na Finish završavaju se inicijalna podešavanja i Eclipse IDE je spreman za razvoj prve Android aplikacije,

Navedeno je prikazano sledećom slikom.

Slika 1.2.6 Application Info

ACTIVITY_MAIN.XML DATOTEKA

Korisnički interfejs aplikacije definisan je activity_main.xml datotekom.

Sada je pažnju neophodno usmeriti na panel pod nazivom Project Explorer U Eclipse integrisanom razvojnom okruženju. U [res/layout](#) folderu dvostrukim klikom bira se datoteka pod imenom [activity_main.xml](#). Ova datoteka definiše korisnički interfejs aplikacije.

Slika 1.2.7 izbor datoteke activity_main.xml

ACTIVITY_MAIN.XML DATOTEKA - PRIMER

Inicijalni kod datoteka moguće je korigovati dodavanjem novih komponentata korisničkog interfejsa.

Inicijalni kod datoteke activity_main.xml sledi ispod:

Slika 1.2.8 activity_main.xml inicijalni kod

Sada je moguće ubaciti i neke vlastite korekcije. Neka to bude još malo teksta i jedno dugme. Ispod prvog xml taga <TextView.../>, a pre završnog taga </LinearLayout>, može se ubaciti sledeći kod:

Slika 1.2.9 Dodavanje koda u xml datoteku

PREVOĐENJE I DEMONSTRACIJA PRIMERA

Nakon snimanja projekta, izborom Run As (Android Application) emulatorom se startuje kreirana aplikacija.

Sada je moguće pristupiti prevođenju i testiranju kreirane aplikacije. Prvo je neophodno snimiti projekat, npr sa ctrl+s, za zatim desnim klikom na naziv projekta selektovati Run As, pa Android Application.

Emulator koji je kreiran simulira rad uređaja na KitKat Androidu i definisan je na način prikazan sledećom slikom.

Slika 1.2.10 Izabrani emulator za testiranje aplikacije

ARHITEKTURA ANDROID APLIKACIJE

Datoteke koje čine hijerarhiju Android projekta prikazane su u Package Exploreru u Eclipse IDE.

Kreirana Android aplikacija će čitaocima ovog materijala, koji se prvi put susreću sa ovom problematikom, biti možda nejasna. Njen osnovni zadatak je, u ovom trenutku, da pokaže početne korake u kreiranju aplikacije – definisanje i podešavanje projekta, kao i hijerarhiju koju je moguće uočiti u Package Explorer-u integrisanog razvojnog okruženja. Budući da postoji više foldera i datoteka koji čine stablo Android projekta, u sledećem izlaganju će biti govora upravo o njihovom sadržaju i nameni.

Za Android projekat od posebnog značaja su sledeći folderi:

- src - sadrži paket sa izvornim datotekama ekstenzije .java. U navedenim datotekama sadržan je kod kojim je omogućeno funkcionisanje posmatrane aplikacije.;
- gen - *sadrži R.java* datoteku. Ovu datoteku je automatski kreirao JAVA prevodilac i ona upućuje na sve resurse u okviru projekta;
- Android 6.0 - folder zavisi od generacije Android SDK kojim se vrši prevođenje, a u ovom slučaju je uključena trenutno najnovija verzija. U svakom slučaju, folder sadrži datoteku android.jar koja predstavlja JAVA arhivu sa svim klasama neophodnim za razvoj neke Android aplikacije;
- assets - folder sadrži sve neohodne alate koje će aplikacija koristiti npr: HTML, tekstualne datoteke, bazu podataka itd;
- bin - folder sadrži sve datoteke koje su nastale angažovanjem ADT u procesu prevođenja aplikacije. Među njima se nalazi i datoteka sa ekstenzijom *.apk (Android Package)* koje predstavlja binarni kod aplikacije i koja se instalira na mobilnom uređaju.
- res - folder u kojem su sadržani svi resursi koji se koriste u aplikaciji;
- AndroidManifest.xml - datoteka o kojoj će biti više govora u kasnijem izlaganju. Osnovna namena joj je definisanje privilegija neophodnih za aplikaciju, kao i filtera sadržaja, primaoca itd.

ORGANIZACIJA PACKAGE EXPLORERA

Svi folderi i datoteke aplikacije dostupni su u Package Exploreru,

Sledećom slikom je prikazan sadržaj projekta u okviru Package Explorera.

Slika 1.2.11 Hijerarhija Android projekta

U daljem izlaganju biće govora o nekim datotekama projekta koje su od posebnog značaja za kreiranje i funkcionisanje aplikacije:

1. activity_main.xml (u starijim verzijama main.xml) datoteka ima za namenu definisanje korisničkog interfejsa datoteke. U prikazanom primeru pokazano je dodavanje tekstualnih komponenata na radnu površinu mobilnog uređaja, kao i kontrole dugme (Button). Posebno bi trebalo napomenuti da je instrukcijom @string/app_name preuzet string definisan u datoteci strings.xml, a koji odgovara nazivu aplikacije. Upravo je i preporuka da se stringovi, koji se koriste u aplikaciji, čuvaju u navedenoj datoteci i da se na njih vrši referenciranje primenom identifikatora @string/*.
2. AndroidManifest.xml je veoma važna datoteka koja sadrži detaljne informacije o aplikaciji kao što su:
3. naziv paketa – u našem slučaju paket ima naziv com.metropolitan.prvaandroidapp;
4. identifikator verzije aplikacije;
5. minimalnu i ciljanu verziju Android OS kojima je aplikacija namenjena;

ORGANIZACIJA PACKAGE EXPLORERA - NASTAVAK

AndroidManifest.xml je datoteka u kojoj su definisane aktivnosti aplikacije.

- aplikacija koristi sliku ic_launcher.png iz drawable foldera;
- android:name=".MainActivity" instrukcijom ukazuje se na aktivnost u aplikaciji;
- Posebnu pažnju, u okviru aktivnosti, trebalo bi obratiti na xml tag <intent-filter> ... </intent-filter> u okviru kojeg se ukazuje na početnu tačku aplikacije (android.intent.action.MAIN), kao i na mogućnost pokretanja aplikacije pomoću launcher ikone(android.intent.category.LAUNCHER).

Sledećim xml kodom prikazan je sadržaj datoteke AndroidManifest.xml sa navedenim informacijama.

Slika 1.2.12 Pogled u AndroidManifest.xml datoteku

R.JAVA DATOTEKA

R.java je automatski ažurirana od strane Eclipse IDE.

Tokom procesa dodavanja datoteka i foldera u Android projekat, datoteka R.java će automatski biti ažurirana od strane Eclipse IDE razvojnog okruženja i nije predviđeno da programer na bilo koji način modifikuje navedenu datoteku.

Slika 1.2.13 Pogled u R.java datoteku

Konačno, datoteka MainActivity.java metodom setContentView() povezuje korisnički interfejs sa aktivnošću.

Slika 1.2.14 Pogled u MainActivity.java datoteku

ZADATAK 1 - OBJASNITE I DOPUNITE KOD ACTIVITY_MAIN.XML

Pokušajte sami

1. Pogledajte kod na slici 9;
2. Objasnite šta će se desiti izvršavanjem ovog koda;
3. Pokušajte da dodate još jednu kontrolu po izboru u *activity_main.xml*, npr. labelu sa vašim imenom ili novo dugme.

▼ Poglavlje 2

Alati i tehnologije za razvoj Android aplikacija

PREGLED NEOPHODNIH ALATA I TEHNOLOGIJA

Programski jezik JAVA je osnovna tehnologija koju koriste alati za razvoj Android aplikacija

U ovom poglavlju biće prikazano preuzimanje i podešavanje sledećih tehnologija i alata neophodnih za razvoj aplikacija za Android operativni sistem:

- JAVA JDK SE;
- Android SDK;
- Eclipse IDE;
- Adroid Development Tools (ADT).

Nakon instalacije navedenih tehnologija i alata biće prikazano kreiranje *Android virtuelnog uređaja (AVD)* kojim će biti omogućeno testiranje kreiranih Android aplikacija.

Sledećom slikom ilustrovane su tehnologije i alati koji će biti preuzeti i instalirani za razvoj Android aplikacija.

Slika 2.1.1 Tehnologije i alati za razvoj Android aplikacija

▼ 2.1 Primer 2 - Podešavanje Jave i Eclipse IDE

JAVA DEVELOPMENT KIT (JDK)

Pre bilo kakvog rada neophodno je prvo instalirati JDK paket.

JAVA Development Kit (JDK) predstavlja implementaciju JAVA platforme predstavljene od strane kompanije *Oracle* u formi paketa binarnih datoteka namenjenih programerima za razvoj JAVA softverskih rešenja na različitim hardversko - softverskim platformama. JDK obuhvata i JAVA virtuelnu mašinu (JVM) i sve prateće resurse kojima je omogućeno razvijanje i izvršavanje JAVA softvera.

Od 2007. godine JAVA se distribuira kroz GNU General Public Licence (GPL), a to znači da je omogućeno potpuno besplatno nabavljanje JAVA tehnologije i odgovarajućih razvojnih alata.

Od 1995. godine, kada je predstavljena prva verzija programskog jezika JAVA, objavljeno je osam generacija ovog programskog jezika zajedno sa tekućim ispravkama.

Sledećom tabelom prikazane su JAVA generacije sa vremenom objavljivanja.

Slika 2.2.1 Generacije JDK paketa

PREUZIMANJE JDK 7

JKD se potpuno besplatno preuzima sa Oracle sajta.

Budući da se JAVA 8 i dalje razvija i dorađuje, JAVA 7 i dalje predstavlja aktuelni standard za razvoj širokog spektra softverskih rešenja. U daljem radu, a sa ciljem formiranja unificiranog alata za razvoj mobilnih aplikacija, prvo će biti demonstrirana instalacija sedme generacije JDK paketa. Prvi korak jeste odlazak na stranicu www.oracle.com i izbor JAVA platforme koja će biti instalirana.

Slika 2.2.2 Izbor JAVA platforme za instalaciju

INSTALACIJA JDK 7

Neophodno je izabrati i instalirati verziju JDK koja odgovara verziji OS instaliranog na računaru.

Nakon izbora platforme, neophodno je da se prihvate uslovi licenciranja softvera i da se izabere verzija JDK u zavisnosti od operativnog sistema na kojem se vrši instalacija, a to je prikazano sledećom slikom.

Slika 2.2.3 Izbor JDK za preuzimanje

Izabrana je instalacija za 64 bitnu verziju operativnog sistema Windows i pokrenuta je instalacija kao što je prikazano sledećom slikom.

Slika 2.2.4 Instalacija JDK

INSTALACIJA JDK 7 - NASTAVAK

Neophodno je izabrati lokaciju i opcione alate za instalaciju.

Tokom procesa instalacije biće neophodno izvršiti određena podešavanja, poput izbora lokacije na kojoj će biti JDK snimljen, kao i izbora opcionih JDK alata.

Slika 2.2.5 Izbor lokacije na kojoj će JDK biti instaliran

Opcioni alati biraju se iz ponuđenih instalacionih menija, instalacija se nastavlja (sledeća slika) i nakon malo vremena dobija se poruka da je instalacija JAVA paketa uspešno obavljena.

Slika 2.2.6 Izbor opcionih JAVA alata

PODEŠAVANJE JDK NA RAČUNARU

Neophodno je JAVU obezbediti dostupnom iz bilo kojeg direktorijuma na računaru.

Sledeći korak je veoma važan, *neophodno je JAVU obezbediti dostupnom iz bilo kojeg direktorijuma na računaru*. To se postiže podešavanjem sistemskih varijabli. Klikom na My Computer, zatim Properties, pa Advanced System Settings i Enviroment Variables, otvara se prozor u kojem će biti izvršena navedena podešavanja. Sada se definiše jedna sistemska promenljiva po imenu JAVA_HOME čija će vrednost biti putanja do JDK foldera. Nakon toga iz liste postojećih varijabli, bira se varijabla PATH u okviru koje je neophodno dodati sledeći string ;%JAVA_HOME%/bin kojim je određena putanja do JAVA prevodioca. Navedeno je prezentovano sledećom slikom.

Slika 2.2.7 Podešavanje JDK sistemskih varijabli

PROVERA KONKRETNOSTI JDK INSTALACIJE.

U MS DOS-u se proverava konkretnost instalacije JAVA paketa.

Na samom kraju neophodno je proveriti da li je JAVA stvarno dostupna iz bilo kojeg foldera. Otvaranjem MS DOS Command Prompt i kucanjem instrukcije *java -version* vrši se provera. Ukoliko se na ekranu ispiše instalirana verzija, zadatak je uspešno obavljen (sledeća slika), u suprotnom biće ispisan komentar 'java' is not recognized as an internal or external command, operable program or batch file.

Slika 2.2.8 Provera uspešnosti JDK instalacije

ANDROID SDK

Android SDK sadrži debager, emulator, dokumentaciju, primere koda i uputstva.

Pored JDK paketa, najvažniji softver koji je neophodno preuzeti je Android SDK. U okviru ovog paketa nalaze se debager, emulator, dokumentacija, primeri koda i uputstva. Odlaskom na web stranicu <http://developer.android.com/intl/vi/sdk/installing/index.html> i izborom opcije *STAND-ALONE SDK TOOLS*, iz menija, *Instaling The SDK*, vrši se preuzimanje instalacionog paketa za Android SDK.

ANDROID SDK - INSTALACIJA

Android SDK može biti preuzet na dva načina: kao .zip paket ili spakovan u Windows Installer.

Sledeće aktivnosti podrazumevaju pokretanje instalacije, izbor lokacije na kojoj će paket Android SDK biti instaliran, kao i obavljanje konačnih podešavanja da bi sve funkcionisalo kako je očekivano. **Android SDK može biti preuzet na dva načina: kao .zip paket ili spakovan u Windows Installer.** U prvom slučaju neophodno je raspakovati arhivu na željenoj lokaciji i izvršiti podešavanja sistemskih varijabli, kao u slučaju sa JAVA JDK. Ako se preuzme Android SDK sa Windows Installerom ova podešavanja će biti izvršena automatski (sledeća slika).

Slika 2.2.9 0 0 Instalacija Android SDK

ANDROID SDK – IZBOR JDK I INSTALACIONOG FOLDERA

Android SDK se vezuje za instalirani JAVA JDK paket.

U sledećem koraku instalacija prepoznaje instaliranu verziju JAVA JDK za koju vezuje instalaciju Android SDK. Klikom na Next instalacija se nastavlja.

Slika 2.2.10 Povezivanje JAVA JDK i Android SDK

Dalje, neophodno je izabrati lokaciju na hard disku gde će biti instaliran Android SDK paket. Izborom lokacije i SDK opcionih alata kompletira se instalacija.

Slika 2.2.11 Izbor lokacije za instalaciju Android SDK

KONFIGURISANJE ANDROID SDK MENADŽERA

Android SDK menadžer upravlja različitim verzijama Android SDK-a instaliranim na računaru.

Pokretanjem instaliranog Android SDK menadžera prikazuje se lista stavki koje su trenutno instalirane na računaru, kao i onih koje su dostupne za naknadnu instalaciju. Izborom iz liste moguće je instalirati dodatne alate, dokumentaciju i platforme koji će biti korišćeni u konkretnim softverskim projektima. Ovde je bitno napomenuti da je svaka verzija Android operativnog sistema određena vlastitim API (Application Programming Interface) nivoom: Android 6 je API nivoa 23, Android 5.1.1 je API nivoa 22, Android 4.4.2 je API nivoa 19 itd. Za svaki API nivo postoje dve platforme:

- **SDK platforma;**
- **Google API interfejs.**

Google API interfejs proširuje SDK funkcionalnostima koji omogućavaju rad sad sa Google servisima, poput Google Maps biblioteka i sl.

Sledećom slikom prikazan je izbor za instalaciju iz Android SDK menadžera. Prihvatanjem uslova licenciranja pokreće se instalacija. Po završetku instalacije biće neophodno restartovati ADB (Android Debug Bridge) što će biti i traženo

Prihvatanjem završava se instalacija izabranih alata.

Slika 2.2.12 Izbor alata iz Android SDK menadžera

ECLIPSE IDE

Eclipse IDE je nazastupljenije razvojno okruženje u industriji softvera.

U daljem radu neophodno je obezbediti razvojno okruženje koje će omogućiti primenu svih navedenih tehnologija i alata. Za razvoj Android aplikacija najčešće se koristi Eclipse IDE (eng. Integrated Development Environment). Eclipse je mnogo više od JAVA IDE – radi se o platformi za razvoj i pokretanje radnih okruženja za različite namene koje nisu ograničene samo na programiranje (mada takvih ima najviše). Najkvalitetnija osobina Eclipse-a jeste proširivost koja se ogleda u preuzimanju različitih dodataka (eng. plug-in) kojima kvalitet razvoja softvera dobija novu dimenziju. Pored podrške JAVA programskom jeziku, Eclipse može da se koristi kao razvojna podrška za druge programske jezike poput: C, C++, Ada, Python itd.

Za razvoj Android aplikacija neophodno je da se sa lokacije eclipse.org/downloads/ preuzme arhiva koja odgovara operativnom sistemu instaliranom na računaru. Jednostavnim raspakivanjem i pozivanjem datoteke eclipse.exe, Eclipse IDE razvojno okruženje se pokreće. Nakon pokretanja biće još neophodno odabrati lokaciju radnog prostora (workspace) gde će biti čuvani svi projekti na kojima će se raditi.

Eclipse je takođe prošao kroz brojne evolucije i ovde će biti govora o najnovijoj verziji Eclipse IDE pod nazivom MARS.

Neophodno je, na kraju, povezati Eclipse sa instaliranim JAVA JDK paketom, navigacijom kroz menije Window – Preferences – Java – Installed JREs (sledeća slika):

Slika 2.2.13 Povezivanje Eclipse IDE sa JDK

ANDROID DEVELOPMENT TOOLS (ADT)

ADT je dodatak za Eclipse IDE za razvoj mobilnih aplikacija.

Da bi bilo omogućeno razvijanje Android mobilnih aplikacija neophodno je instaliranim Eclipse IDE razvojnom okruženju priključiti dodatak pod imenom [Android Development Tools](#). Ovim proširenjem omogućen je razvoj mobilnih aplikacija ali i kvalitetnije identifikovanje i otklanjanje grešaka nastalih tokom procesa programiranja. Dodatak ADT omogućava da se kroz Eclipse IDE uradi neki od sledećih zadataka:

- Kreiranje novog Android projekta;
- Upotreba Android emulatora;
- Upravljanje greškama;

- Preslikavanje Android aplikacija u Android pakete (APK);
- Kreiranje digitalnih sertifikata APK paketa.

Instalacija ADT je jednostavna. U Eclipse IDE bira se meni Help, a zatim se bira opcija Install New Software. U ponuđenom okviru za dijalog neophodno je uneti sledeći string <https://dl-ssl.google.com/android/eclipse> koji odgovara URL-u sa kojeg se poziva instalacija ADT. Klikom na dugme Next pokreće se instalacija i ADT se integriše u Eclipse IDE.

Navedeno je ilustrovano sledećom slikom.

Slika 2.2.14 Instalacija ADT za Eclipse IDE

ECLIPSE I ANDROID - VIDEO MATERIJALI

Pregled instalacije Eclipse IDE i neophodnih tehnologija kroz video materijale

Instalacija Jave

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

Instalacija Eclipse IDE

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

ZADATAK 2 - POKUŠAJTE SAMOSTALNO DA PODESITE ECLIPSE IDE

Samostalno podešavanje Eclipse razvojnog okruženja

Budući da ste kroz prethodne predmete već instalirali i podesili Javu, pokušajte samostalno i sledeće:

1. Instalirajte, i podesite za razvoj Android aplikacija, na vašem računaru razvojno okruženje *Eclipse IDE*;
2. Podesite u razvojnom okruženju *Android Development Tools (ADT)*;
3. Kreirajte *AVD* u *Eclipse IDE* za testiranje kreiranih Android aplikacija.

▼ 2.2 Primer 3-Instalacija i podešavanje Android Studio

SISTEMSKI ZAHTEVI

Pre instalacije neophodno je proveriti da li razvojni svojim hardverskim resursima dozvoljava instalaciju Android Studija.

Pre bilo kakvog rada neophodno je obezbediti da se računar ponaša kao Android razvojna platforma. To podrazumeva instalaciju Java razvojnog alata – JDK (videti lekciju 1) i instalacionog paketa *Android Studio* koji uključuje *Android SDK*. Ukoliko je *Android SDK* već instaliran, budući da je već korišćen za Eclipse IDE, dovoljno je instalirati *Android Studio* bez *SDK*.

Da bi *Android Studio* mogao da bude instaliran na konkretan računar, neophodno je da budu podržani sledeći zahtevi:

- Windows 2003 (32-bit ili 64-bit)
- Windows Vista (32-bit ili 64-bit)
- Windows 7 (32-bit ili 64-bit)
- Windows 8 / Windows 8.1
- Windows 10
- Mac OS X 10.8.5 or later (Intel based systems only)
- Linux systems with version 2.11 or later of GNU C Library (glibc)
- Minimum of 2GB of RAM (4GB is preferred)
- 1.5GB of available disk space

PREUZIMANJE ANDROID STUDIO PAKETA

Android Studio paket se slobodno preuzima sa odgovarajuće web lokacije.

Budući da je u Lekciji 1 bilo detaljno govora o instalaciji JDK i proveri ispravnosti instalacije, u ovoj lekciji neće biti govora o tome. Akcenat će biti stavljen na preuzimanje i instalaciju paketa *Android Studio*. Instalacioni paket je moguće preuzeti sa sledeće lokacije: <http://developer.android.com/sdk/index.html>.

Odlaskom na ovu stranicu bira se instalacioni paket koji odgovara operativnom sistemu koji je instaliran na ciljnom računaru (sledeća slika).

Slika 2.3.1 Preuzimanje Android Studio paketa

INSTALIRANJE ANDROID STUDIO PAKETA

Nakon preuzimanja pokreće se instalacija i locira se Java folder.

U folderu u kojem je snimljen preuzeti paket, pronalazi se datoteka pod opštim nazivom **android-studio-bundle-<version>.exe** i dvostrukim klikom miša pokreće se instalacija. Otvara se prozor dobrodošlice i klikom na *Next* proces se nastavlja.

Slika 2.3.2 Početak instalacije

U nastavku, neophodno je navesti putanju do JDK instalacionog foldera na računaru (sledeće slike).

Slika 2.3.3 Izbor instalacionog JDK foldera

ANDROID STUDIO PAKET – INSTALACIONI FOLDER

U daljem radu se definiše lokacija na kojoj će Android Studio biti instaliran.

Nakon identifikovanja instaliranog Java JDK-a, tokom instalacije, pristupa se definisanju instalacionog foldera za Android Studio IDE. Pre finaliziranja ove aktivnosti neophodno je definisati komponente koje će biti instalirane na računar (sledeća slika).

Slika 2.3.4 Izbor komponentata za instalaciju

Ukoliko već postoji SDK na računaru, nema potrebe za ponovnim instaliranjem. Klikom na *Next* otvara se novi prozor u kojem će biti definisan instalacioni folder za Android Paket i prateće komponente (sledeća slika).

Slika 2.3.5 Izbor instalacionog foldera

SPECIFICIRANJE RAM MEMORIJE ZA EMULACIJU I KRAJ INSTALACIJE

U nastavku izvode se završna podešavanja instalacionog paketa.

U nastavku, u narednom prozoru, određuje se RAM memorija koja se rezerviše za Android emulator. Navedeno je prikazano sledećom slikom.

Slika 2.3.6 Rezervisanje RAM-a za emulaciju

Klikom na *Next* počinje raspakivanje Android Studio paketa sa izabranim podešavanjima i komponentama. Kada se raspakuju sve datoteke, klikom na dugme *Finish* završava se instalacija Android Studio paketa.

Slika 2.3.7 Kraj instalacije

ZADATAK 3- POKUŠAJTE SAMOSTALNO DA PODESITE ANDROID STUDIO

Samostalno podešavanje razvojnog okruženja Android Studio IDE

Budući da ste kroz prethodne predmete već instalirali i podesili Javu, pokušajte samostalno i sledeće:

1. Instalirajte, i podesite za razvoj Android aplikacija, na vašem računaru razvojno okruženje *Android Studio IDE*;
2. Kreirajte *AVD*-a za testiranje aplikacija kreiranih u *Android Studio IDE*

▼ 2.3 Primer 4 - Pokretanje Android Studio alata

PRVO POKRETANJE ANDROID STUDIO IDE ALATA

Nakon završetka instalacije Android Studio se pokreće po prvi put.

Nakon završetka instalacije Android Studio se pokreće po prvi put. Na naslovnoj strani Android Studio IDE alata nudi se nekoliko opcija:

- Izbor novog Android projekta;
- Otvaranje postojećeg Android projekta;
- Uvoz postojećih Android primera;
- Provere projekata;
- Uvoz projekata iz drugih IDE (Eclipse itd);
- Podešavanja, i
- Dokumentacija

Navedeno je prikazano sledećom slikom.

Slika 2.4.1 Početna strana

KREIRANJE NOVOG PROJEKTA

Klikom na Start new Android Studio Project započinje se razvoj novog Android projekta.

Na naslovnoj strani, pokrenutog Android Studio IDE alata, izborom opcije **Start new Android Studio Project** započinje se razvoj novog Android projekta. Podešavanja parametara novog projekta podrazumevaju definisanje naziva aplikacije i paketa u kojem će se aplikacija čuvati.

Takođe, ovde je neophodno definisati i lokaciju na hard disku na kojoj će folder projekta biti snimljen.

Navedeno je prikazano sledećom slikom.

Slika 2.4.2 Izbor naziva aplikacije, paketa i lokacije paketa

IZBOR ANDROID API VERZIJE I ŠABLONA KLASE AKTIVNOSTI

Prilikom razvoja nove aplikacije moguće je izabrati neki od ponuđenih šablona i verziju Android API.

Nakon izbora navedenih podešavanja na red dolazi izbor uređaja kojem je aplikacija namenjena (mobilni telefon ili tablet, TV i Wear) kao i minimalnog SDK za koji se aplikacija razvija. Navedeno je prikazano sledećom slikom.

Slika 2.4.3 Min SDK i podržani Android uređaji

Još jedna aktivnost se izvodi pre samog prelaska na programiranje Android aplikacije, a to je izbor od nekih ponuđenih šablona za klasu aktivnosti. Moguće je izabrati praznu klasu aktivnosti ili neku od predefinisanih klasa sa specifičnim namerama (sledeća slika).

Slika 2.4.4 Šabloni klase aktivnosti

KREIRANJE NOVOG PROJEKTA - KRAJ

Izvođenjem prethodnih koraka sve je spremno za razvoj nove Android aplikacije.

Izborom nekog od navedenih šablona za klasu aktivnosti i klikom na *Finish* završili smo inicijalna podešavanja za buduću aplikaciju i sve je spremno za programiranje (sledeća slika). U nastavku je neophodno kreirati odgovarajući emulator (Android Virtual Device – AVD) za testiranje aplikacije koja se kreira.

Slika 2.4.5 Podešen projekat za programiranje

Sledećim video materijalom pokazan je razvoj jednostavne Android aplikacije u razvojnom okruženju Android Studio IDE.

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

ZADATAK 4 - PRVI SAMOSTALNI ANDROID STUDIO PROJEKAT

Pokušajte sami

Pokušajte, vodeći se aktuelnim primerom, da samostalno kreirate vlastiti Android Studio projekat.

▼ 2.4 Primer 5 -Kreiranje emulatora u Android Studio IDE

KREIRANJE AVD U ANDROID STUDIJU

Emulator se kreira na skoro identičan način kao u Eclipse IDE.

U Android Studio IDE okruženju bira se opcija **AVD Manager** kojom je omogućeno kreiranje novog Android emulatora. Navedeno je pokazano sledećom slikom.

Slika 2.5.1 Pokretanje AVD Menadžera

PODEŠAVANJE AVD U ANDROID STUDIJU

Neophodno je podesiti AVD uređaj za buduće testiranje Android aplikacija.

U prozoru *Android Virtual Device* omogućeno je pokretanje postojećeg emulatora, njegovo modifikovanje ili kreiranje novog. Klikom na opciju *Create Virtual Device* pokreće se novi prozor za kreiranje i podešavanje novog emulatora. Navedeno je pokazano sledećom slikom.

Slika 2.5.2 Virtual Device Manager

U novom prozoru moguće je izabrati neki od predefinisanih emulatora ili kreirati vlastiti. Takođe, moguće je izabrati jednu od sledećih kategorija Android uređaja koji se emuliraju: telefon, tablet, TV ili Gear (sledeća slika).

Slika 2.5.3 Izbor hardvera za emulaciju

PODEŠAVANJE AVD U ANDROID STUDIJU – NOVI UREĐAJ I API VERZIJA

*Izborom opcije **New Hardware Profile** definiše se novi AVD uređaj.*

Da bi bilo omogućeno kreiranje novog AVD uređaja, neophodno je, u nastavku, izabrati opciju *New Hardware Profile*. Ova opcija daje mogućnost definisanja raznih osobina emulatora poput: veličine ekrana, omogućavanja kamere, žiroskopa i akceleratora, izbora orijentacije ekrana itd. Navedeno je pokazano sledećom slikom.

Slika 2.5.4 Nov hardver za emulaciju

Klikom na OK, kontrola se vraća na prethodni prozor u kojem se sada nalazi kreirani emulator kao postojeći harvdver. U narednom koraku, izborom opcije *Next*, bira se dopunsko podešavanje kreiranog emulatora gde se definiše Android API za kreirani emulator (sledeća slika).

Slika 2.5.5 Izbor API verzije za emulator

PODEŠAVANJE AVD U ANDROID STUDIJU – KRAJ DEFINICIJE

*U prozoru **Verify Configuration** obavlja se završetak definicije kreiranog AVD.*

Konačno, u prozoru *Verify Configuration* obavljaju se završna podešavanja kreiranog AVD uređaja i prihvatanje svih dotadašnjih podešavanja. Klikom na dugme *Finish* biće završena definicija željenog AVD uređaja (sledeća slika).

Slika 2.5.6 Kraj definisanja AVD

Emulator je kreiran i spreman za pokretanje što se vidi sledećom slikom.

Slika 2.5.7 Prozor za pokretanje kreiranog emulatora

Sledeći video materijal pokazuje instalaciju i podešavanje Android Studio IDE okruženja i zaokružuje prethodno izlaganje.

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

ZADATAK 5- PRVI SAMOSTALNI ANDROID STUDIO EMULATOR

Pokušajte sami

Kreirajte i podesite emulator za testiranje aplikacija iz *Android Studio IDE*.

Preporuka: Kreirajte emulator koji simulira realni Android uređaj sa najnovijom *API* verzijom.

▼ 2.5 Primer 6 - Importovanje projekata iz Eclipse IDE

EKSPORTOVANJE PROJEKTA IZ ECLIPSE IDE

Prvi korak je priprema projekta za eksportovanje iz Eclipse u Android Studio.

Da bi projekat, kreiran u Eclipse IDE, uspešno bio uključen u Android Studio, neophodno je prvo izvršiti njegovu pripremu za eksportovanje iz Eclipse IDE okruženja. U Eclipse meniju bira se opcija *File* i u okviru nje opcija *Export* (sledeća slika).

Slika 2.6.1 Priprema projekta za eksport iz Eclipse IDE

U sledećem koraku neophodno je kreirati *Gradle Build Files* kojima će biti omogućeno eksportovanje datoteka iz Eclipse IDE u Android Studio (sledeća slika).

Slika 2.6.2 Generisanje Gradle Build datoteka

IMPORTOVANJE PROJEKTA U ANDROID STUDIO

Nakon kreiranja Gradle Build datoteka, projekat je moguće prebaciti u Android Studio okruženje.

Nakon kreiranja Gradle Build datoteka, projekat je moguće prebaciti u Android Studio okruženje. U prvom koraku se iz Android Studio menija bira opcija *File*, iz koje se biraju podopcije *New* i *Import Project* (sledeća slika).

Slika 2.6.3 Početak uvoza projekta u Android Studio

U sledećem koraku se bira konkretan projekat koji se uvozi u Android Studio, navigacijom do njegove lokacije na disku (sledeća slika).

Slika 2.6.4 Izbor projekta za importovanje

IMPORTOVANJE PROJEKTA U ANDROID STUDIO - KRAJ

U nastavku Android Studio vrši analizu datoteka koje uvozi.

Nakon izbora projekta i klika na dugme *OK*, Android Studio započinje procesiranje uvoza izabranih datoteka. Ceo proces je javan i vidljiv korisniku i počinje sekvencom sa sledeće slike.

Slika 2.6.5 Kreiranje Gradle Project informacija

Ceo proces će potrajati neko vreme, nakon čega će korisnik da dobije informaciju o uspešno obavljenom zadatku (sledeća slika).

Slika 2.6.6 Kraj procesa - kreiranje Gradle Project informacija

Na samom kraju, projekat je uspešno importovan u Android Studio i spreman je za dalju obradu (sledeća slika).

Slika 2.6.7 Importovan projekat u Android Studio

Sledećim video materijalom pokazano je kako može Eclipse projekat biti preseljen u Android Studio.

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

ZADATAK 6 - NASTAVITE RAZVOJ ECLIPSE PROJEKTA U ANDROID STUDIO IDE

Pokušajte sami

Kreirajte jednostavnu Android aplikaciju u *Eclipse* razvojnom okruženju.

Eksportujte kreiranu aplikaciju u *Android Studio IDE*;

Dodajte novu funkcionalnost po izboru;

Pokrenite i testirajte aplikaciju u *Android Studio IDE*.

▼ Poglavlje 3

Android emulator Genymotion

EMULATOR VISOKIH PERFORMANSI

Emulator visokih performansi podiže kvalitet testiranja Android aplikacija.

U dosadašnjem izlaganju prikazano je kako se kreira AVD instanca u reprezentativnim Android razvojnim okruženjima Eclipse IDE i Android Studio IDE. Ovakvi emulatori su uzimali značajnu količinu RAM memorije od računara na kojem su instalirane, imali su problema sa brzinom otvaranja i reagovanja na postavljene zadatke testiranja kreiranih Android aplikacija. Problem je dodatno izražen kod PC računara koje pokreće AMD procesor. Ovi procesori, za razliku od Intel procesora, ne podržavaju hardversko ubrzavanje emulacije i taj proces deluje još sporiji i frustrirajući.

Problem je moguće prevazići korišćenjem emulatora visokih performansi koji je podržan od Android razvojnih okruženja. Primer ovakvog emulatora je [Genymotion](https://www.genymotion.com/) kojeg je moguće preuzeti sa linka <https://www.genymotion.com/>.

Slika 3.1.1 Početna stranica za inszalaciju Gennymotion emulatora

GENYMOTION EMULATOR

Za funkcionisanje Gennymotion emulatora biće instaliran i Oracle VM Virtual Box.

Sama instalacija emulatora je veoma jednostavna. Preuzme se instalaciona datoteka sa navedene adrese i prati se uputstvo za instalaciju. Zbog trivijalnosti ovog zadatka, lekcija mu neće pridavati posebnu pažnju. Ono što je neophodno istaći, a za potrebe funkcionisanja Genymotion emulatora, tokom instalacije biće instaliran i **OracleVM Virtual Box** kao neophodna podrška za rad ovog emulatora.

Slika 3.1.2 Oracle VM Virtual Box

Nakon uspešno obavljene instalacije celokupnog instalacionog paketa, moguće je pokrenuti po prvi put Genymotion i izvršiti njegovo povezivanje sa odgovarajućim razvojnim alatom instalacijom odgovarajućih dodataka (*plug-in*-ova). Za Android Studio, moguće je preuzeti dodatak za Gennymotion emulator na stranici <https://www.genymotion.com/plugins/> i pratiti jasna uputstva za instalaciju ovog dodatka.

Slika 3.1.3 Gennymotion plug-in

U sledećem video materijalu dat je lep prikaz ovog emulatora.

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

PRIMER 7 - GENYMOTION EMULATOR - POKRETANJE

Genymotion nudi veliki broj unapred definisanih mobilnih uređaja za testiranje aplikacija.

Nakon uspešne instalacije i povezivanja sa razvojnim okruženjem, moguće je pokrenuti Genymotion emulator i vršiti testiranje kreiranih Android aplikacija pomoću njega.

Slika 3.1.4 Gennymotion početni prozor

Pored opcija za pokretanje emulatora (**znak za PLAY**) i za podešavanja (**Settings**), na slici je moguće primetiti i opciju **Add** kojom se bira željena platforma emulatora. Genymotion nudi veliki broj unapred definisanih mobilnih uređaja za testiranje aplikacija. Svaku od ovih instanci moguće je redefinisati i prilagoditi vlastitim željama i potrebama.

Slika 3.1.5 Izbor ponuđene AVD instance

GENYMOTION EMULATOR - POKRETANJE AVD INSTANCE

Klikom na izabranu instancu emulatora, pokreće se konkretan emulator.

Klikom na izabranu instancu emulatora, pokreće se konkretan emulator za testiranje Android mobilnih aplikacija.

Slika 3.1.6 Pokretanje Gennymotion instance

U veoma kratkom roku, brže nego i jedan drugi emulator, pojavljuje se izabrana instanca emulatora za testiranje aplikacija.

Slika 3.1.7 Gennymotion AVD instanca

Sada je moguće testirati kreirane Android aplikacije.

GENYMOTION EMULATOR - TESTIRANJE KREIRANE APLIKACIJE

Izborom Genymotion emulatora iz ponuđene liste, testira se aplikacija ovim AVD-om.

Aplikaciju kreiranu u razvojnom okruženju, na primer Android Studio, sada je moguće testirati ovim emulatorom. Ako se pretpostavi da je otvoren konkretan Android projekat u Android Studio IDE razvojnom okruženju, klikom na **Run Application** otvoriće se prozor za izbor emulatora koji se želi izvršiti testiranje aplikacije.

Slika 3.1.8 Pokretanje aplikacije

Slika 3.1.9 Izbor iz liste dostupnih emulatora

Iz liste dostupnih emulatora bira se Genymotion i aplikacija se prosleđuje na testiranje instanci ovog emulatora.

ZADATAK 7 - TESTIRAJTE VLASTITU APLIKACIJU POMOĆU GENYMOTION AVD

Pokušajte sami

Vašu aplikaciju, koju ste prethodno kreirali, pokrenite u emulatoru *Genymotion AVD*.

▼ 3.1 Primer 8 - Razvoj aplikacije u Android Studiju

VEŽBA 1 - NOV PROJEKAT, KREIRANJE JEDNOSTAVNE APLIKACIJE

Iz menija Android Studio IDE bira se opcija File-New-New Project za kreiranje novog Android projekta.

Pokretanjem alata Android Studio i izborom opcije menija *File-New-New Project* započinje se kreiranje novog projekta. U konkretnom slučaju aplikacija će nositi naziv *AndroidStudioProba*. Definisanje naziva aplikacije, paketa i odredišnog foldera prikazano je sledećom slikom.

Slika 3.2.1 Otvaranje novog projekta

IZBOR CILJNOG ANDROID UREĐAJA

U sledećem koraku definiše se uređaj i API na kojem će se aplikacija izvršavati.

Sledeći, obavezan, korak u razvoju novog Android projekta jeste definisanje uređaja kojem je aplikacija namenjena i, potom, izbor minimalnog SDK koji će podržati izvršavanje aplikacije. U konkretnom slučaju, aplikacija će biti namenjena telefonima i tabletima, a minimalni podržani SDK će biti *API 15 (Android 4.0.3. IceCreamSandwich)*. Sledeća slika ilustruje navedeno.

Slika 3.2.2 Izbor uređaja i minimalnog SDK

IZBOR ŠABLONA AKTIVNOSTI

U sledećem koraku biće dodata aktivnost u aplikaciju.

U nastavku definisanja projekta dodaje se aktivnost u aplikaciju. Ovde se sreće nekoliko šablona za kreiranje glavne aktivnosti, a u konkretnom slučaju biće izabrana prazna aktivnost (Empty Activity) koja će u daljem radu biti redefinisana. Sledećom slikom biće prikazano navedeno. U nastavku, biće prihvaćeni ponuđeni nazivi za klasu aktivnosti i XML GUI datoteku.

Slika 3.2.3 Izbor šablona aktivnosti

DODAVANJE KONTROLA U GUI

Kreiranu datoteku korisničkog interfejsa neophodno je redefinisati konkretnim pogledima.

U folderu app, hijerarhije projekta, fokus će biti stavljen na podfolder res i njegovu datoteku *activity_main.xml*. Kao što je već poznato, radi se o datoteci korisničkog interfejsa. U konkretnom slučaju biće redefinisana da pokazuje poglede u linearnom vertikalnom redosledu. Pored *TextView* pogleda biće dodato i jedno dugme čijim klikom će biti izvršena konkretna metoda klase aktivnosti. Kod *activity_main.xml* datoteke, dat je sledećom slikom.

Slika 3.2.4 Redefinisanje activity:main.xml datoteke

KREIRANJE KLASE AKTIVNOSTI I TEST PRIMERA

U sledećem koraku biće finiširan jednostavni primer primene Android Studio IDE.

Završna definicija ovog jednostavnog primera biće data kroz redefinisane inicijalne klase aktivnosti. Za ugrađeno dugme biće dodat osluškivač i metoda koja će se izvršiti kada se desi događaj klik na dugme. Kod je dat sledećom slikom.

Slika 3.2.5 Klasa aktivnosti kreirana Android Studio IDE alatom

Pokretanjem aplikacije u kreiranom emulatoru moguće je testirati valjanost kreirane aplikacije (sledeća slika).

Slika 3.2.6 Test primera u emulatoru

ZADATAK 8 - KREIRAJTE NOVU ANDROID APLIKACIJU

Vežbanje kreiranja aplikacija primenom Android Studio IDE.

Na vašem računaru obaviti i dokumentovati, a vodeći se instrukcijama iz ovog dela lekcije:

1. Podešavanje i instalaciju Gennymotion AVD-a za testiranje aplikacija kreiranih u Android Studio IDE;
2. Kreirati nov Android projekat;
3. Kreirati i pokrenuti Gennymotion ili neku drugu AVD instancu.
4. Dokumentovati izvršavanje aplikacije u AVD instanci.

Obavljene zadatke dokumentovati odovarajućim slikama ekrana (screenshot).

▼ Poglavlje 4

Domaći zadatak 1

ZADATAK 1– KREIRANJE PROJEKTA I JEDNOSTAVNA IZMENA GUI

Kreiranje novog projekta, dodavanje komponenata u GUI, prevođenje i testiranje.

1. Pokrenuti Android Studio IDE i kreirati nov projekat. Projektu i aplikaciji dodeliti naziv Student1, za paket izabrati com.metropolitan.student1;
2. Za SDK prihvatiti ponuđena podešavanja;
3. Promeniti naziv glavne aktivnosti u Student1Activity, a za layout naziv zadržati activity_main.
4. Otvoriti datoteku activity.main.xml i promeniti izgled iz RelativeLayout u LinearLayout. Takođe, podesiti da LinearLayout ima vertikalni raspored;
5. U ponuđenom <TextView ... /> tagu promeniti podrazumevanu vrednost stringa @string/hello_world tako da on nakon promene prikazuje naziv aplikacije;
6. Definirati dva nova dugmeta sa natpisima OK i Cancel raspoređena jedan ispod drugog, a ispod gore navedenog teksta.
7. Snimiti izmene u activity.main.xml
8. Kreirati AVD emulator po sopstvenoj želji primenom Android Studio IDE ili Gennymotion;
9. Pokrenuti kreiranu virtuelni uređaj;
10. Prevesti aplikaciju Student1 i pokrenuti je na AVD emulatoru.
11. U slučaju grešaka otkloniti ih i ponoviti pod 10. U suprotnom, kopirati datoteku Student1.apk na telefon, instalirati je i pokrenuti.

▼ Zaključak

PREGLED LEKCIJE L01

Lekcijom01 dat je pregled Android OS sa akcentom na njegove mogućnosti.

U prethodnom izlaganju bilo je govora o osnovama Android operativnog sistema sa osvrtom na aktuelnu i prethodne verzije. Posebno je bilo govora o specifičnim karakteristikama ovog operativnog sistema, kao i alatima i tehnologijama neophodnim za razvijanje aplikacija namenjenih ovom operativnom sistemu.

Pokazano je preuzimanje i podešavanje JAVA JDK paketa bez kojeg rad na izradi Android aplikacija ne bi bio moguć, a zatim je demonstrirana instalacija razvojnog okruženja Eclipse IDE kojim su svi alati i tehnologije za razvoj mobilnih aplikacija objedinjeni. Dalje, preuzet je i podešen alat Android SDK koji sadrži debager, emulator, dokumentaciju, primere koda i uputstva neophodne za razvoja Android aplikacija. U nastavku, SDK menadžerom su izabrane i instalirane API datoteke koje će biti korišćene, a takođe, demonstrirana je primena Android virtuelnog uređaja za testiranje kreiranih aplikacija.

Dalje, prezentovano je još jedno razvojno okruženje Android Studio IDE sa svim podešavanjima za Javu i SDK neophodnim za razvoj Android aplikacija primenom ovog razvojnog okruženja.

Na kraju, kreirana je prva Android aplikacija sa ciljem pokazivanja koraka u kreiranju i izvršavanju android projekta, kao i arhitekture Android aplikacije. Prilikom testiranja aplikacije, korišćen je emulator visokih performansi Gennymotion.

LITERATURA

Za pripremanje lekcije korišćena je sledeća literatura

1. Lee W. M. 2012. *Android 4 – razvoj aplikacija*, Wiley Publishing, INC
2. <http://developer.android.com/training/index.html>
3. <http://www.tutorialspoint.com/android/>
4. <http://www.vogella.com/tutorials/android.html>
5. <https://www.genymotion.com>
6. <https://developer.android.com/studio/install.html>