Android

Lectures on University College
for Applied Computer Engineering
http://www.racunarstvo.hr/en/

Miljenko Cvjetko
http://holisticware.net

Android - Introduction

Goals

- understand fundamentals of developing Android applications
    - programming tool installation,
    - project creation and coding, and
    - running an app on a virtual (emulator) or physical Android device
- gain knowledge
    - how to tools use
    - application fundamentals
    - components

Android - Introduction

Requirements

- programming (native)
  - java knowledge
    - at least reading and comprehension
  - optional
    - Xamarin: C# and F#

Android - Introduction

Labs (Hands on)

- https://sites.google.com/site/androidappcoursev3/
- https://sites.google.com/site/androidappcourse/labs/
- http://androidlabs.org/

Android - Basics

Definition

http://www.android.com/

- Android definition
  - Linux
  - +
  - Java API
- hardware
  - ARM

    ABIs
    - ARM 5
    - ARM 7
  - x86
  - MIPS

Android - Basics
Market share and Monetization

- market share
  - most popular mobile OS
  - 2014-01-29
    80%
    http://www.engadget.com/2014/01/29/strategy-analytics-2013-smartphone-share/
- Monetization
  - low
  - competition - huge
  - fragmentation
  - large numbers, but small to none profit

Android - Basics

Installation of Development Environment

01

Android - Basics

Installation of Development Environment 02

- Eclipse ADT Plugin (Android Development Toolkit)
- Android SDK
  - SDK Manager
  - AVD Android Virtual Device (Emulator) Manager android virtual machine for emulator
    - API version (OS)
    - form factor (size)
    - Memory, SD card...

Android - Basics

Installation of Development Environment 03

- Device Monitor
  Dalvik Debug Monitor Server (DDMS)
  - System Information, FileSystem, Memory, processor, Threads, Network
  - Emulator Manager
  - Call and Message simulation - incomming
  - Location simulation
  - Android Platforms - Versions
    - http://developer.android.com/about/index.html
    - http://developer.android.com/about/dashboards/index.html

    fragmentation!!!
  - Packages

Android - Basics

Installation of Development Environment 04

- Emulator
  - google emulator with SDK
    Google Intel x86 HAXM driver
- ADB Android Debug Bridge
  command line utility
  bridges IDE and device (real or virtual)
  in ./platform-tools/
  bridge between Eclipse (user) and Android SDK and device
- Extras
  - Emulator: GenyMotion
    - does not work in Virtual Machines

Android - Tools

Projects 01

1. Eclipse
2. File +/ New +/ Project
3. +/ Android
4. +/ Android Application project
5. Application Name
6. Project Name
7. Package Name = Namespace
   tld.company.appname

Android - Tools
Projects 02

1. Minimum Required SDK
   - cannot run on lower levels
   - lower levels - more devices
   - default - enables targeting majority of platform versions
2. Target SDK
   - default - latest version
   - preferred that user chooses the lowest build number possible
   - compatible with the largest number of existing systems in place
3. Compile with
   - set this to be equal as Target SDK
4. library - reusable component

Android - Project Structure

Project structure 01

- src/
  - java sources (Activities, Services ...)
- bin/
  - files that are automatically generated by the compiler and toolchain
  - code and resources and packaged application itself
  - .apk file
    zip file
- libs/
  - location of all JAR files for additional features and functionality support in Android programming
  - for support libraries
  - android-support-v4.jar

Android - Project Structure

Project structure 02

Android - Project Structure

Project structure 03

- res/
  - folder will contains resources
  - external data files
  - naming constraints
    - lower case letters
    - the numbers 0-9
    - special characters underscores '_', or periods '.' (single!!)
  - three main types of resources in different subfolders
    - layout/
    - drawable-*/
    - values-*/

Android - Project Structure

Project structure 04

- res/
  - res/drawble-x/
    - folders contain image and animation files used in application.
    - several of these folders for various screen sizes/densities
    - each folder contains default files
      - icon.png
        application icon after installation
      - ic_action_search.png
        search features

Android - Project Structure

Project structure 05

- res/
  - res/layout/
    - XML layouts *.xml (*.axml) files
    - user interfaces
  - res/menu/
    - hold XML menu files
    - menus (Menu button) for various app options when app is in foreground
  - res/values/
    - XML files that contain value type resources (strings, integers)

Android - Project Structure

Project structure 06

Android - Project Structure

Project structure 07

- assets/
  - similar to resources
  - for asset files
  - anything stored is accessed in the classic file manipulation style
  - AssetManager class is used to open the file, read in a stream of bytes, and process the data

Android - Project Structure

Project structure 08

- ./ (root folder of the project)
  - AndroidManifest.xml
    information about application
    - Package name for identification
    - app name
    - version
    - list of components used
      (activities, services, broadcast recievers, content providers)
    - permissions app needs to run (internet access, location ...)
    - profiling information
    - libraries, external components
    - API levels

Android - Project Structure

Project structure 09

- ./ (root folder of the project)
  - proguard-project.txt
    obfuscation, shrinking
  - project.properties
    - project settings
    - targets (build
    - do not edit manually - do it in IDE - eclipse

Android - Concepts
Application Elements/Components 01

- programmatic concepts
  java classes/objects
  - Activities + Intents
    - similar to a web page,
    - basic building blocks
    - usually display (load) a user interface
    - single interactive screen
    - can be w/o UI
    - Intents - Navigation
  - Services
    - run in background and perform task
    - usually have no UI
    - setup and launched by activities

Android - Concepts

Application Elements/Components 02

Android - Concepts

Application Elements/Components 02

- programmatic concepts
  java classes/objects
  - UI
    - defined
      - declarative in XML
        res/layout/*.xml
        Xamarin *.axml files
      - programmatically in code

Android - Concepts

Application Elements/Components 03

Android - Concepts

Application Elements/Components 04

- programmatic concepts
  java classes/objects
  - UI -
    - Views (Widgets, Controls, UIViews)
      - single (simple)
        single element (inherits from View)
        - Button
        - TextView (Label, UILabel)
        - EditText (TextBox, UITextField, UITextView)
        - CheckBox
        - ...
      - complex (groups of views - ViewGroup)

Android - Concepts

Application Elements/Components 05

- programmatic concepts
  java classes/objects
    - UI -
        - Views (Widgets, Controls, UIViews)
            - complex (groups of views - ViewGroup)
              layouts
                - AdapterViews (ListView)
                - pickers
                - Layouts - inherit from ViewGroup

Android - Concepts

Application Elements/Components 06

- Layouts 01
  - AbsoluteLayout

    absolute (x,y) coordinate position for each child view. This is a rigid layout that pins all child views down exactly where they are specified. Using this does not allow the user interface to adjust for different screen sizes and resolutions.
  - LinearLayout

    mostly used

    list of child views and draws them sequentially in a single direction, either horizontally or vertically. You have the option of assigning a weight value for each child view which determines how much it is allowed to grow if there is extra space.
  - FrameLayout

    Takes a single view object and simply pins it to the upper left hand corner of the screen. Each child view that is added is drawn on top of the previous one, causing it to be completely or partially obscured without very careful screen size and view hierarchy arrangement.
  - GridLayout

Android - Concepts

Application Elements/Components 07

- Layouts 02
  - RelativeLayout
    'responsive' design
    Positions its child views relative to other child views or the parent view. For example, two child views can be left-justified in the parent, or one child view can be made to be below another.
  - TableLayout
    'responsive' design
    Positions its child views in a grid of rows and columns. A row is a child view specified by the TableRow class. TableRows can have zero or more cells and can contain empty cells. However, a cell cannot span multiple columns. Each cell is itself another view, like a Button, and can be set to shrink or grow.

Deploying

- device (real or virtual)
- app must be digitally signed
- no need for certificate authority (Verisign)
- can be self signed for debug builds
- release and Google play
- no debug certificate

Testing Debugging

- log cat
- DDMS

Projects - Ideas

1 /3 = 33 points
8 points

- Calendar
  - multiple
    users calendars / topics / categories
  - chooser
  - day / week / month
- Timer / Alarm
  start, stop, pause. countdown, from file
- Messaging
  send SMS, Mail
  subject, Content / Body, recipients
- MultiMedia player
  Images, slideshow, Audio, Video
- Voice Commander