

MIT Inventor

prvý dotyk aplikácie

Peter Borovanský
KAI, I-18

borovan 'at' ii.fmph.uniba.sk



Ako začať s App Inventor

<http://appinventor.mit.edu/explore/ai2/setup-emulator.html>

- potrebujete google-mail účet
- <http://ai2.appinventor.mit.edu/>

Počítač:

- platformy: MS-Windows, Mac OS X, Ubuntu, Debian
- browser: FF, Safari, Chrome, ~~IE~~
- setup page: <http://appinventor.mit.edu/explore/ai2/setup.html>
- pre MS-Windows treba pustiť **MIT Appinventor Tools 2.3.0 (~80 MB)**



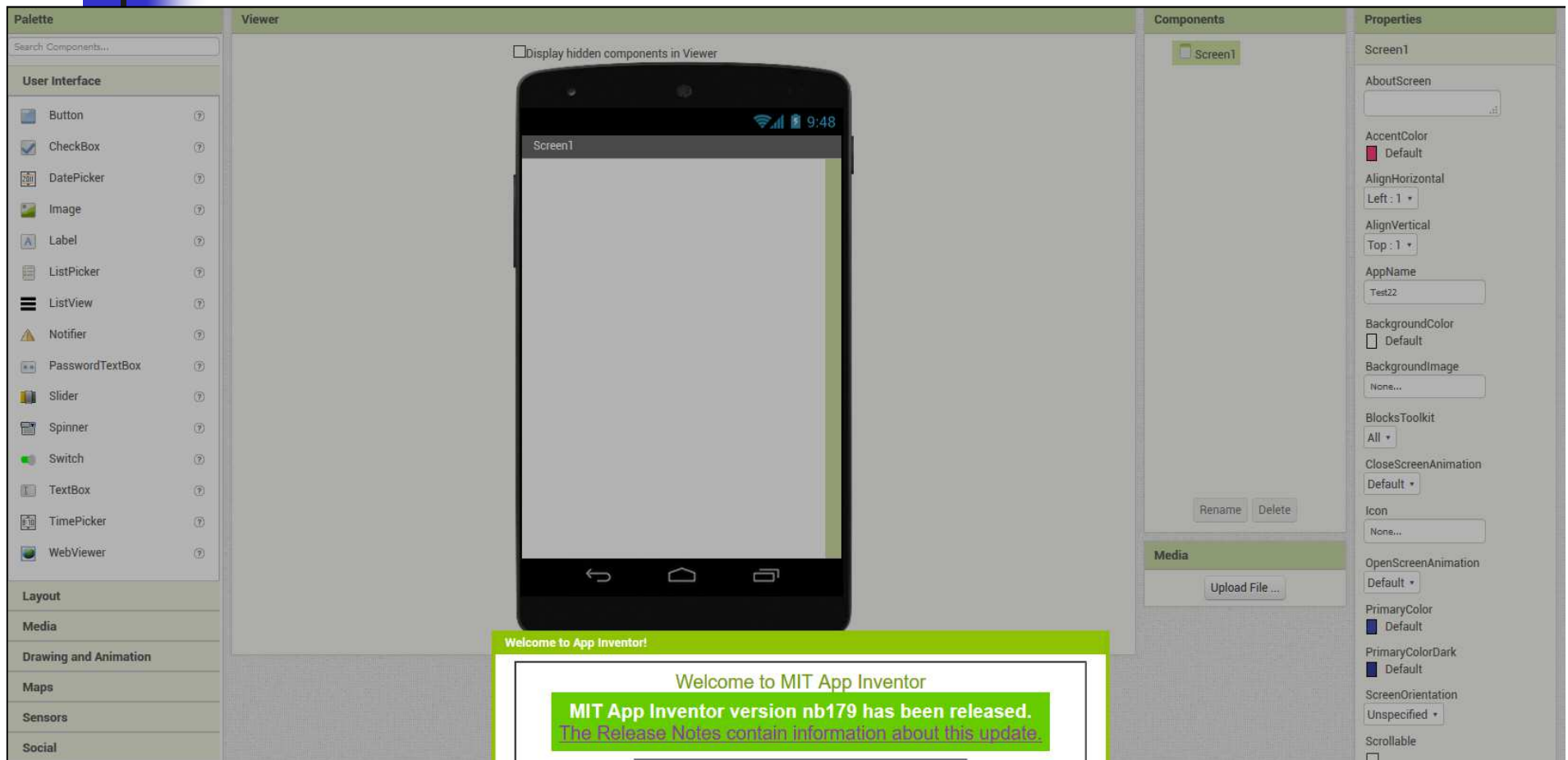
Na zariadení:

<https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>

Android Mobil (ale prežijete aj s emulátorom):

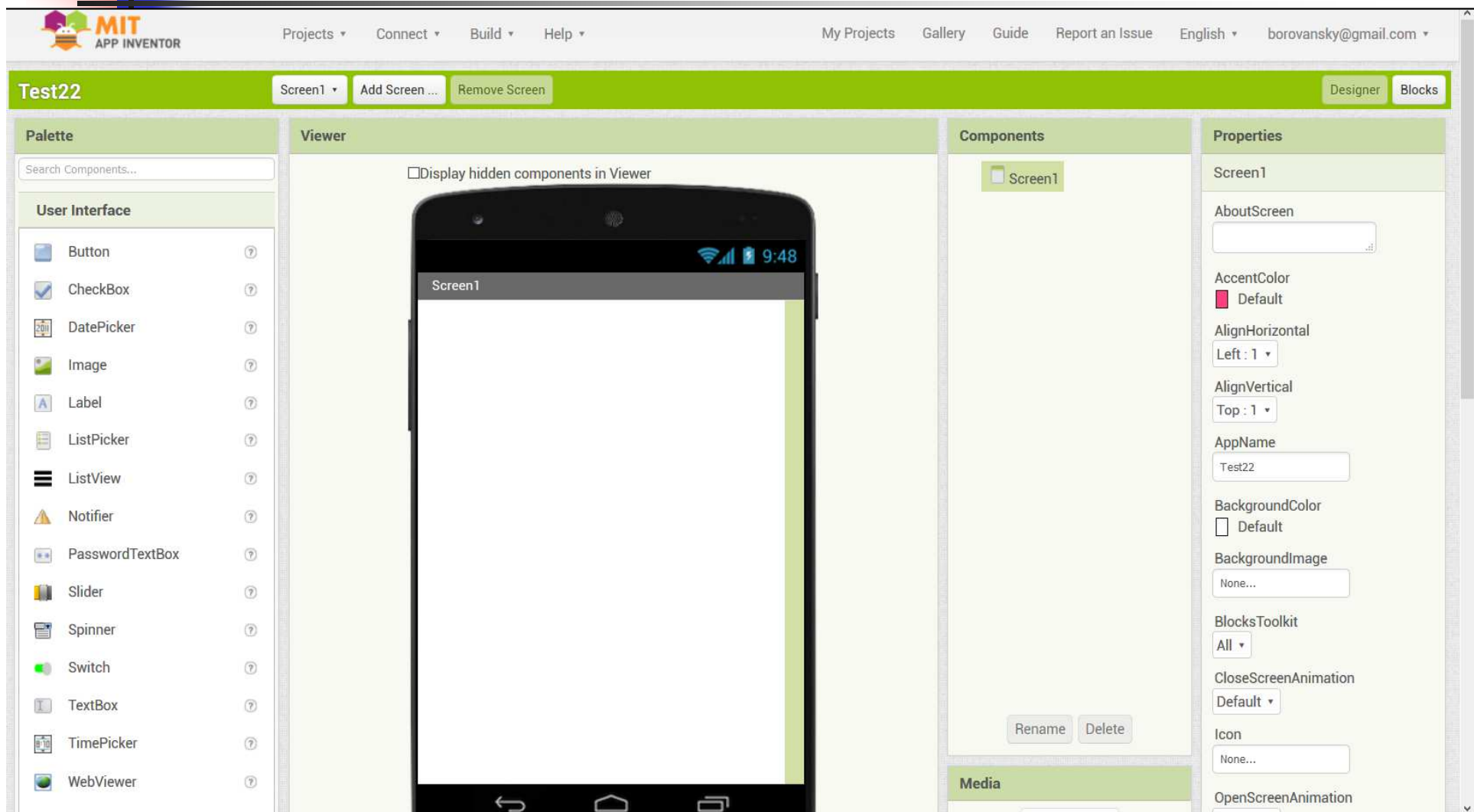
- v mobile: Setting/Application/Development/, 7xklik na Build Number
USB Debugging = ON (môžete uploadovať vlastné aplikácie .apk)
Stay awake = ON (nebude vám usávať, kým ho máte na kábli)
Allow mock location = ON (ak chcete používať nejaké fake GPS – neskôr)
Sound & Display/Orientation = OFF

MIT Inventor – prvý pohľad



<http://ai2.appinventor.mit.edu/>

MIT Inventor – hlavní panel



Android Apps with App Inventor: The Fast and Easy Way to Build Android Apps

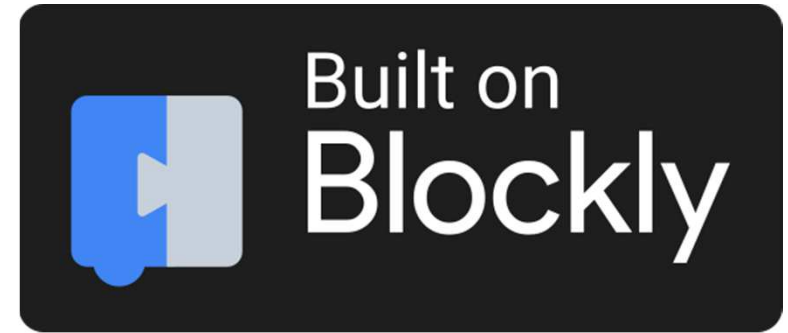
J.H.Kloss, Amazon: 4.4/5

1. Installing and configuring App Inventor
2. Building modern, attractive mobile user interfaces
3. Controlling Android media hardware, including the camera
4. Saving data locally with TinyDB, or in the cloud with Tiny
5. Streamlining and automating phone, text, and email com
6. Tracking orientation, acceleration, and geoposition
7. Integrating text-to-speech and speech-to-text in your apps
8. Controlling other apps and Web services with ActivityStarter
9. Building mobile mashups by exchanging data with Web APIs
10. Testing your apps for diverse hardware with the Android Emulator
11. Example apps

<http://books.google.sk/books?id=rfKyt6TRLloC&pg=SA4-PA7&lpg=SA4-PA7&dq=laughter+android+kloss&source=bl&ots=Kht9zKTIyK&sig=YqHJbc8v77VtR28RmpZccyV2ScY&hl=sk&sa=X&ei=bGOZUMmgDMbesqbA64HIBq&ved=0CCkQ6AEwAQ#v=onepage&q&f=true>

http://www.amazon.com/Android-Apps-App-Inventor-Build/dp/0321812700/ref=sr_1_7?s=books&ie=UTF8&qid=1322522259&sr=1-7





Iné zdroje

- AppInventor at MIT (<http://appinventor.mit.edu/>)
- Kurz edX: Mobile Computing with App Inventor - CS Principles (english)
(<https://courses.edx.org/courses/course-v1:TrinityX+T007x+1T2017/course/>)
- Kurz Learn2Code: MIT Inventor (slovensky)
(<https://www.learn2code.sk/kurzy/mit-app-inventor>)
- príklady hotových projektov, ak radšej čítate hotové projekty:
App Inventor Snippets (<http://puravidaapps.com/snippets.php>)
- App Inventor Teach – pre učiteľov (<http://appinventor.mit.edu/explore/teach.html>)
- MIT Inventor Tutorials – Hour of Code (<http://appinventor.mit.edu/explore/hour-of-code.html>)
- MIT Inventor Public Open Source (<http://appinventor.mit.edu/appinventor-sources/>)
- Dlho očakávané MIT Inventor for iOS (<http://doesappinventorrunonios.com/>)

My projects

zoznam
mojich
projektov

import
export

Connect

Build

login

The screenshot shows the MIT App Inventor web interface. The browser address bar displays `ai2.appinventor.mit.edu/?locale=en#568460635170`. The top navigation bar includes links for Projects, Connect, Build, Help, My Projects, Gallery, Guide, Report an Issue, English, and a user profile for borovansky@gmail.com. The main content area is divided into two sections: 'My Projects' on the left and a table of projects on the right.

My Projects List:

Name	Date Modified	Published
<input type="checkbox"/> ShockMe	Sep 29, 2015, 8:42:36 PM	No
<input type="checkbox"/> PresporksePivociary	Sep 17, 2015, 7:47:00 PM	No
<input type="checkbox"/> demo_Media	Oct 1, 2016, 1:23:14 PM	No
<input type="checkbox"/> Prvy	Oct 1, 2016, 1:08:03 PM	No
<input type="checkbox"/> Labilo	Sep 17, 2015, 7:45:04 PM	No
<input type="checkbox"/> MazeLabyrinth	Sep 30, 2016, 12:30:40 PM	No
<input type="checkbox"/> Dynamic		
<input type="checkbox"/> SpiritLevel		
<input type="checkbox"/> Xxxx		
<input type="checkbox"/> Twiitingo		
<input type="checkbox"/> FireBaseDemo		
<input type="checkbox"/> EV3Robot		
<input type="checkbox"/> Social		
<input type="checkbox"/> next		
<input type="checkbox"/> EV3		
<input type="checkbox"/> BaseEV3Project		
<input type="checkbox"/> Hallooooo		
<input type="checkbox"/> PokusPrednaska		

My projects menu:

- Start new project
- Import project (.aia) from my computer ...
- Import project (.aia) from a repository ...
- Delete Project
- Save project
- Save project as ...
- Checkpoint
- Export selected project (.aia) to my computer
- Export all projects
- Import keystore
- Export keystore
- Delete keystore

Connect menu:

- AI Companion
- Emulator
- USB
- Reset Connection
- Hard Reset

Build menu:

- App (provide QR code for .apk)
- App (save .apk to my computer)

Prvý projekt

The screenshot displays the MIT App Inventor web interface. At the top, the MIT App Inventor logo is on the left, and navigation links for Projects, Connect, Build, Help, My Projects, Gallery, Guide, Report an Issue, English, and a user email (borovansky@gmail.com) are on the right. Below the navigation bar, a green header bar contains the project name 'Prvy2017', a dropdown menu for 'Screen1', and buttons for 'Add Screen ...' and 'Remove Screen'. On the far right of this bar are 'Designer' and 'Blocks' tabs.

The main workspace is divided into four panels:

- Palette:** A list of components categorized into User Interface, Layout, Media, Drawing and Animation, Sensors, Social, Storage, and Connectivity. Under Connectivity, there are icons and names for ActivityStarter, BluetoothClient, BluetoothServer, and Web. Below these are sections for LEGO MINDSTORMS, Experimental, and Extension.
- Viewer:** A central area showing a preview of the app. It includes checkboxes for 'Display hidden components in Viewer' and 'Check to see Preview on Tablet size.'. Below these is a simulated mobile screen showing a status bar with Wi-Fi, cellular signal, and the time 9:48. The screen content is labeled 'Screen1'.
- Components:** A panel showing a list of components added to the app, currently containing 'Screen1'.
- Properties:** A panel showing the properties for the selected component, 'Screen1'. It includes fields for 'AboutScreen', 'AlignHorizontal' (set to 'Left : 1'), 'AlignVertical' (set to 'Top : 1'), 'AppName' (set to 'Prvy2017'), 'BackgroundColor' (set to 'White'), 'BackgroundImage' (set to 'None...'), 'CloseScreenAnimation' (set to 'Default'), 'Icon' (set to 'None...'), and 'OpenScreenAnimation' (set to 'Default').

Connect WiFi

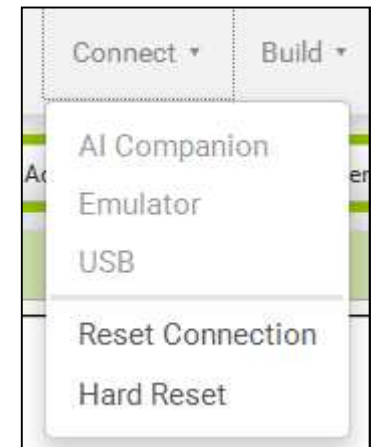
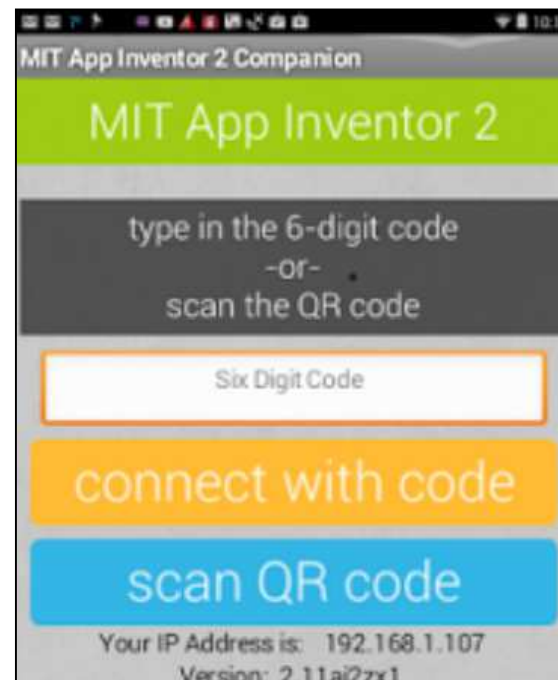
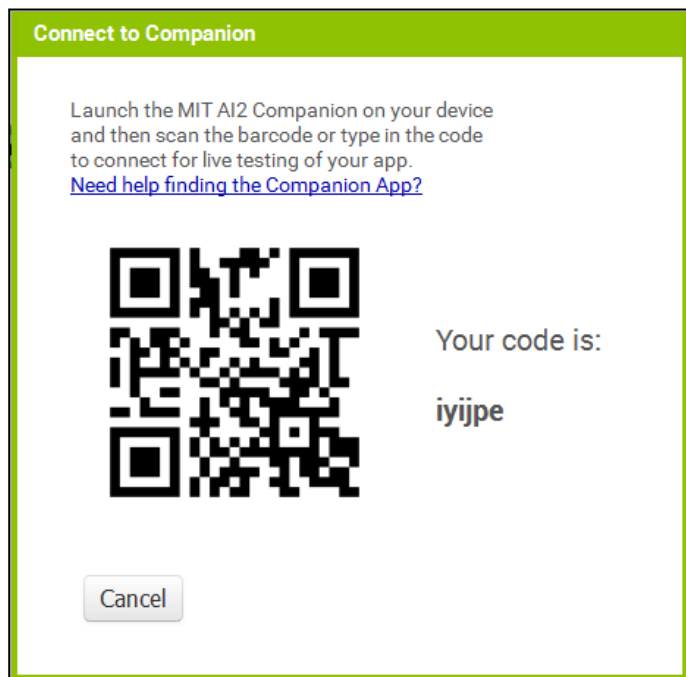


- Nainštalujte MIT AI2 Companion App cez Google Play Store

<https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>

Connect to Device

- spusti MIT AI2 Companion App



Palety komponentov

- User Interface

- Button, CheckBox, Clock, Image, Label, List/Time/Date-Picker, Password, Slider, TextBox

- Layout

- Horizontal/Vertical/Table Arrangement [Scrollable]

- Media

- Camcoder, Camera, ImagePicked, Player, Sound, VideoPlayer, TextToSpeech, SpeechRecognizer

- Drawing and Animation

- Ball, Canvas, ImageSprite

- Social

- ContactPicker, EmailPicker, PhoneNumberPicker, PhoneCall, Texting, Twitter, Sharing

- Sensor

- Accelerometer, Location, Orientation, Gyro, Pedometer, Proximity, Bar Code Scanner

- Storage

- File, TinyDB, FireBaseDB

- Connectivity

- BluetoothClient-Server, ActivityStarter

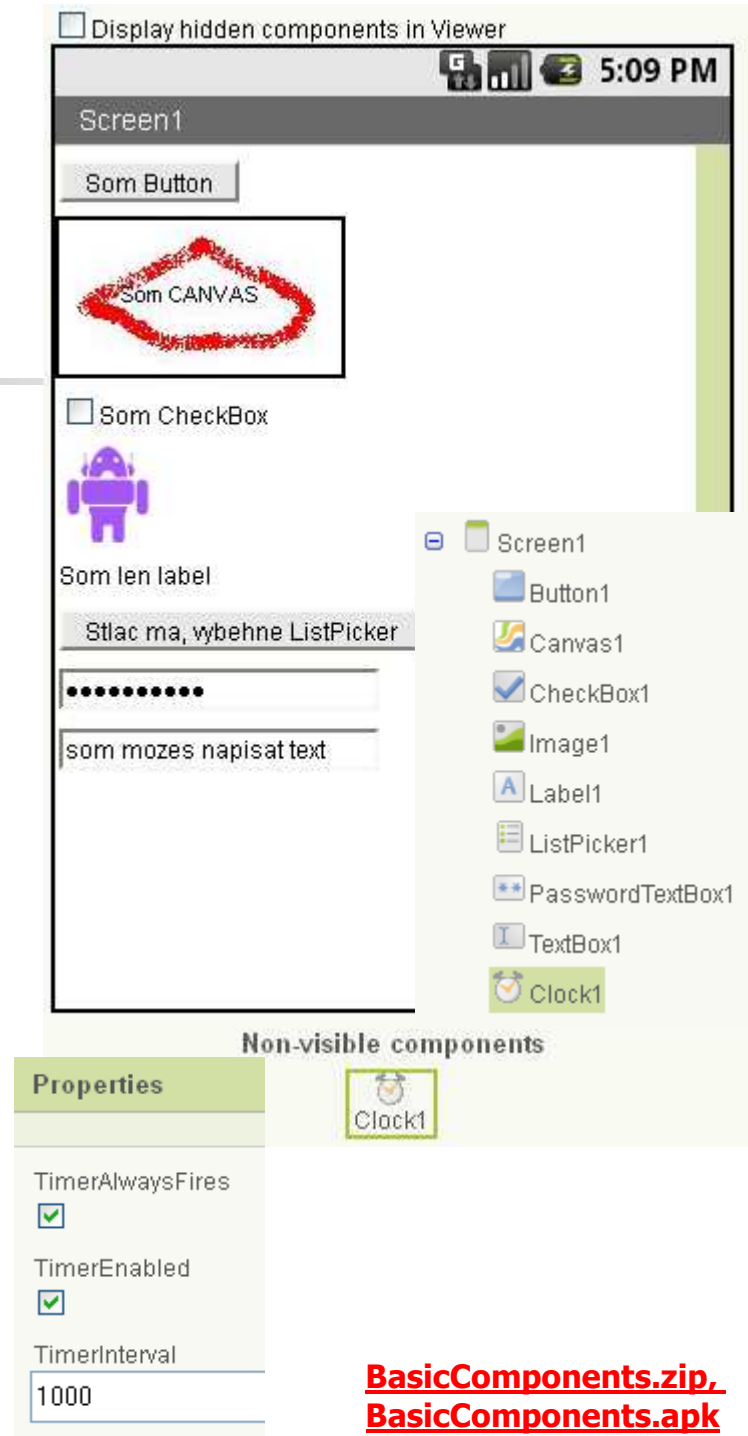
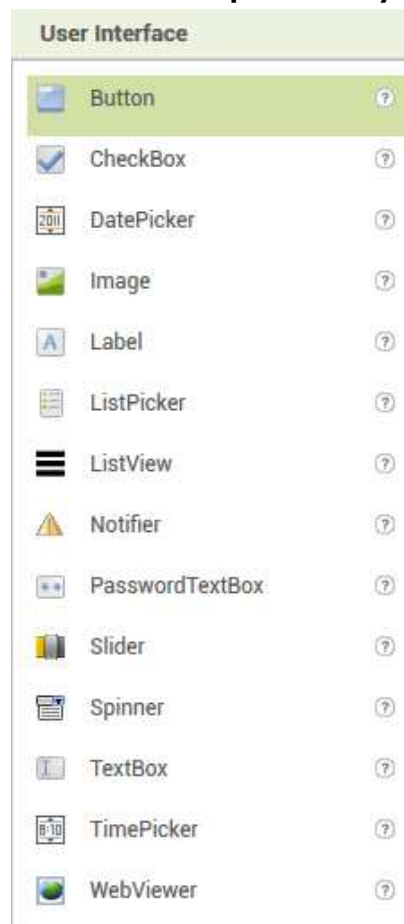
- Lego MINDSTORMS



User Interface

Ilustrujeme si základné vizuálne komponenty:

- [Button](#)
- [CheckBox](#)
- [Clock](#)
- [Image](#)
- [Label](#)
- [List/Time/Date-Picker](#)
- [PasswordTextBox](#)
- [TextBox](#)
- Notifier
- WebViewer



Layouts (Arrangement)

- HorizontalArrangement
- TableArrangement
- VerticalArrangement

☐ Display hidden components in Viewer

Screen2

Tah: Cas:

4	3	5	1
7	9	12	2
6	10	13	.
8	11	14	15

Start Stop

Layout

- HorizontalArrangement ?
- HorizontalScrollArrangement ?
- TableArrangement ?
- VerticalArrangement ?
- VerticalScrollArrangement ?

Properties

Columns

4

Rows

4

Visible

showing ▼

Width

Fill parent...

Height

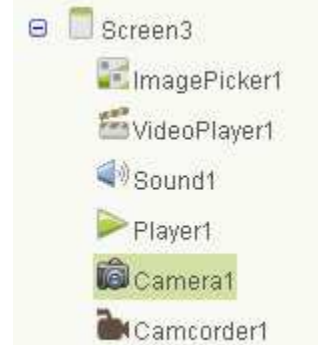
Fill parent...

- Screen2
 - HorizontalArrangement1
 - Label1
 - TextBox1
 - Label2
 - TextBox2
 - TableArrangement1
 - Button2
 - Button3
 - Button4
 - Button5

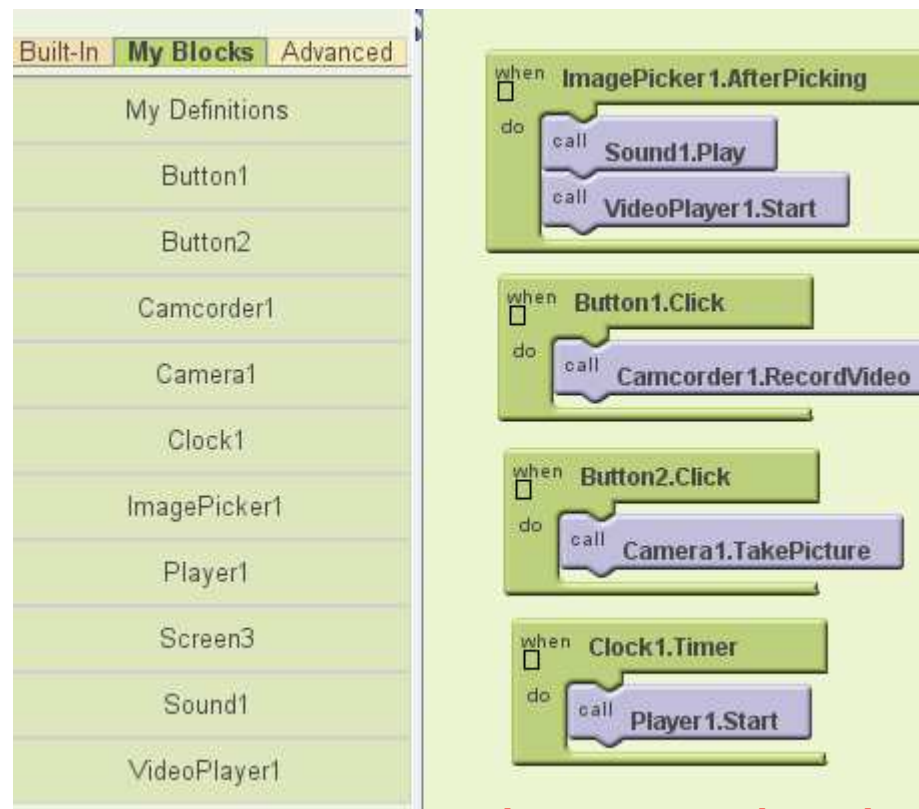
Media

- Camera
- ImagePicker
- Player
- Sound
- VideoPlayer

- Spustiť: block editor
- Uložiť (strýčko Google si to ukladá sám v MyProjects:-)

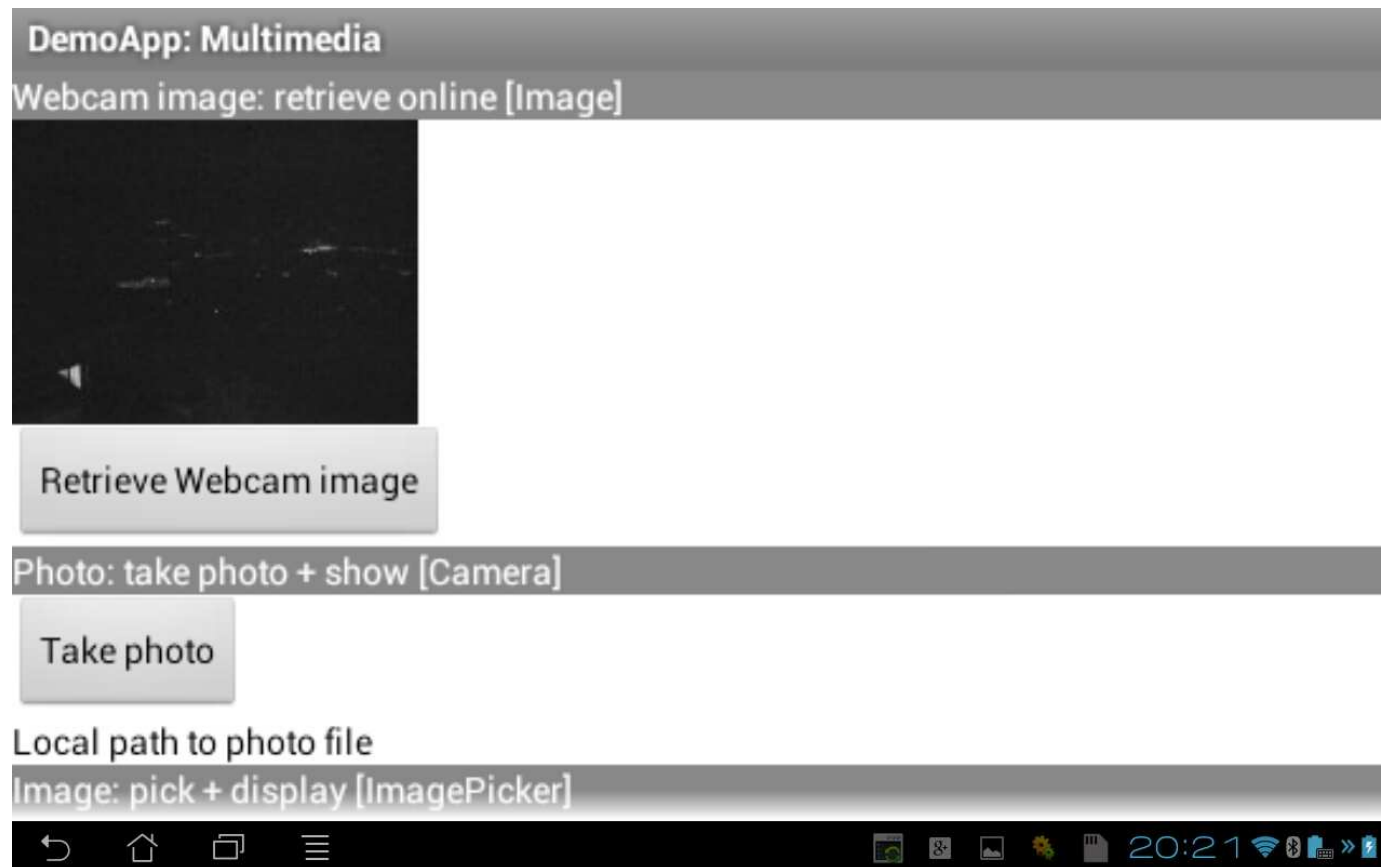


Non-visible components



[BasicComponents.zip](#), [BasicComponents.apk](#)

demo_Media



FingerPaint1

jednoduché malovátko prstom



```
when BtnBlue.Click
do set Canvas1.PaintColor to blue

when BtnGreen.Click
do set Canvas1.PaintColor to green

when BtnRed.Click
do set Canvas1.PaintColor to red

when ButtonClear.Click
do call Canvas1.Clear

when ButtonSmall.Click
do set Canvas1.LineWidth to 5

when ButtonBig.Click
do set Canvas1.LineWidth to 15
```

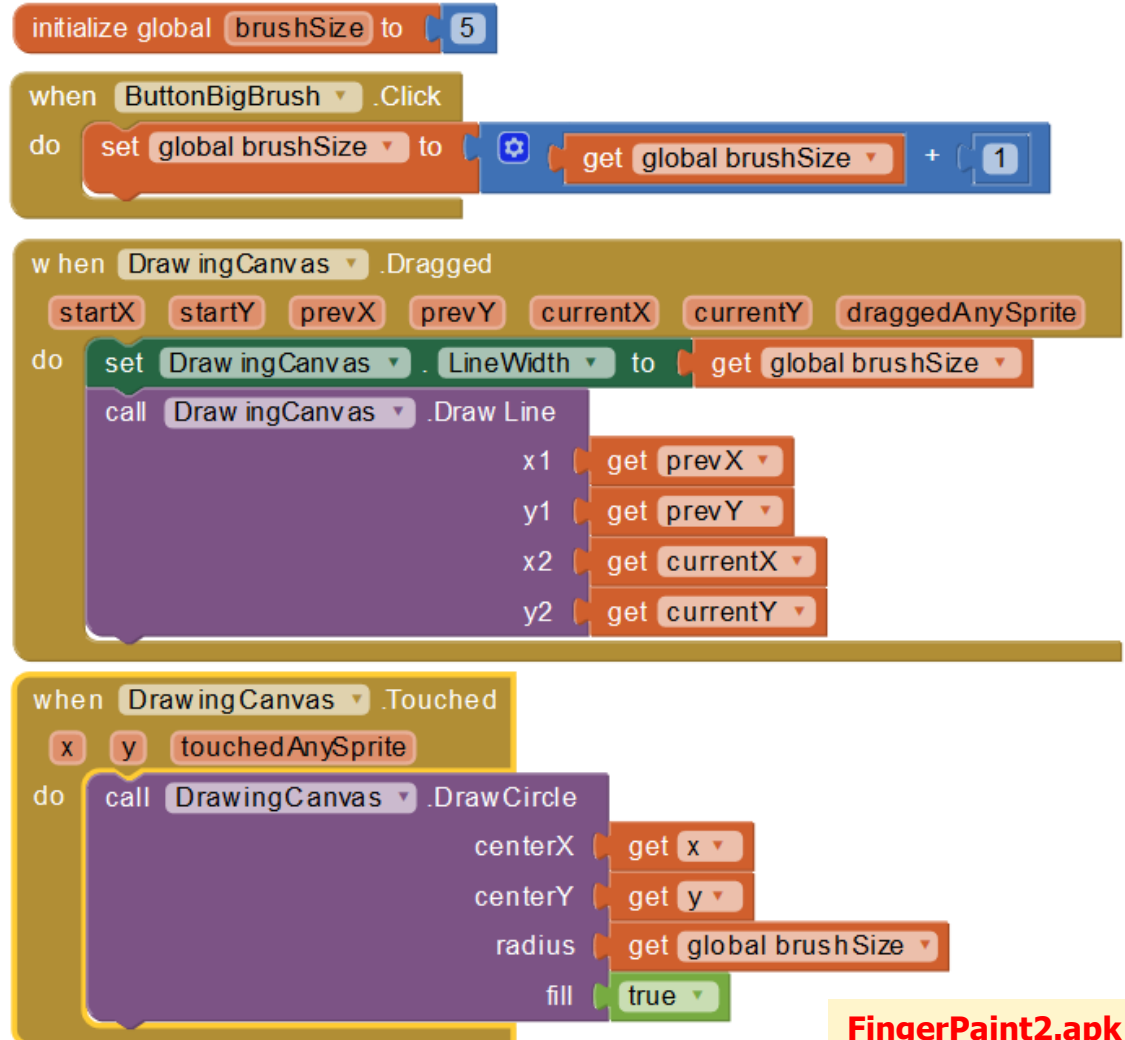
```
when Canvas1.Touched
  x y touchedAnySprite
do call Canvas1.DrawCircle
  centerX get x
  centerY get y
  radius 10
  fill true

when Canvas1.Dragged
  startX startY prevX prevY currentX currentY draggedAnySprite
do call Canvas1.DrawLine
  x1 get prevX
  y1 get prevY
  x2 get currentX
  y2 get currentY
  Draw a line on the screen.
```


FingerPaint2

Pokročilejšia verzia

- Hrúbka pera
 - globálna premenná brushSize
- Kreslenie čiar
 - drawingCanvas.Dragged



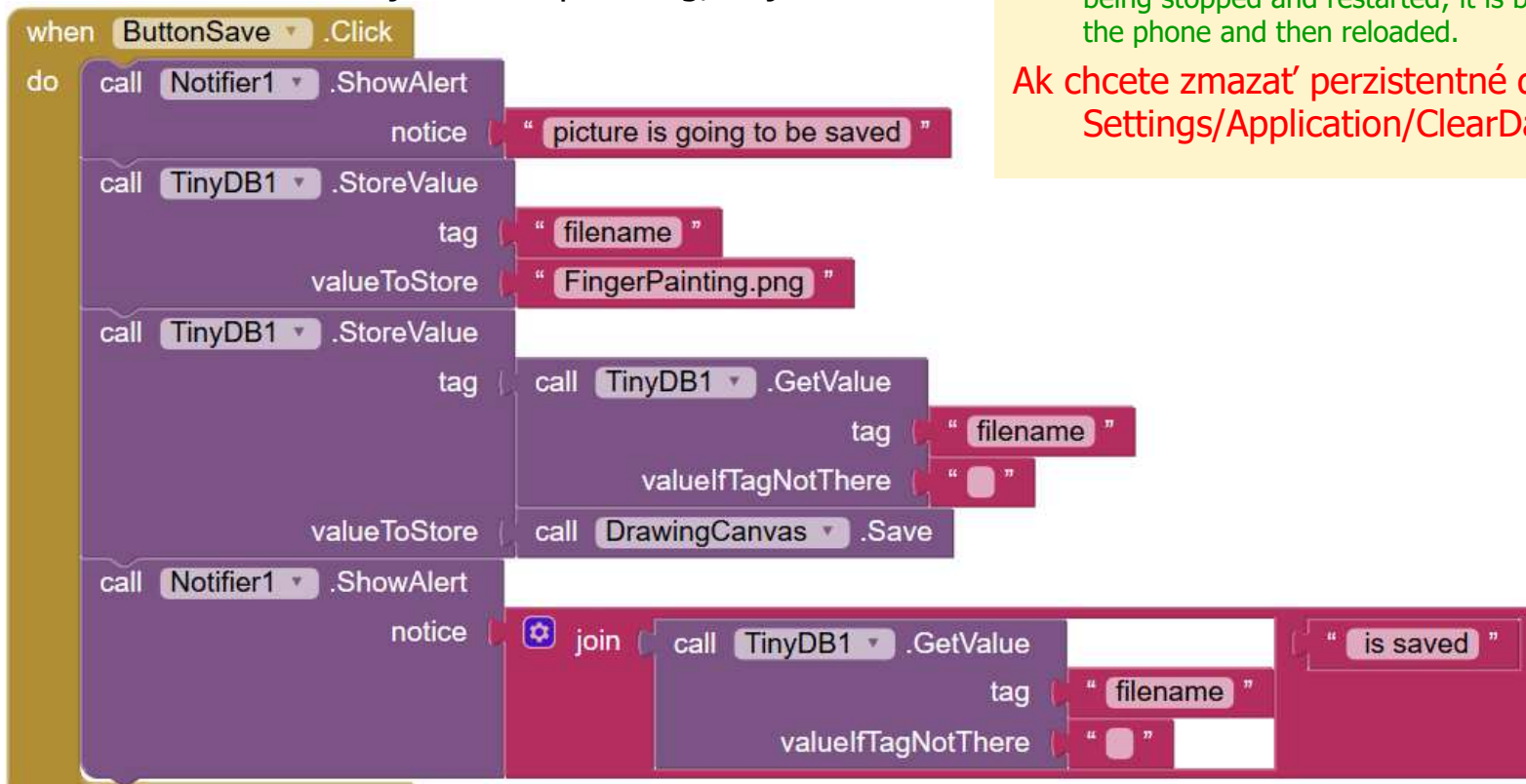
[FingerPaint2.apk](#)

FingerPaint2

- Ukladanie obrázku
 - Persistencia
- Práca s TinyDB
 - čo je HashMap<String, Object>

The data in TinyDB is persistent only when you have packaged and downloading your app. If you are developing connected to the phone, and you restart the Appinventor application, or if you disconnect and reconnect the phone, then the data base will start fresh. This is a case where the application is not merely being stopped and restarted; it is being removed from the phone and then reloaded.

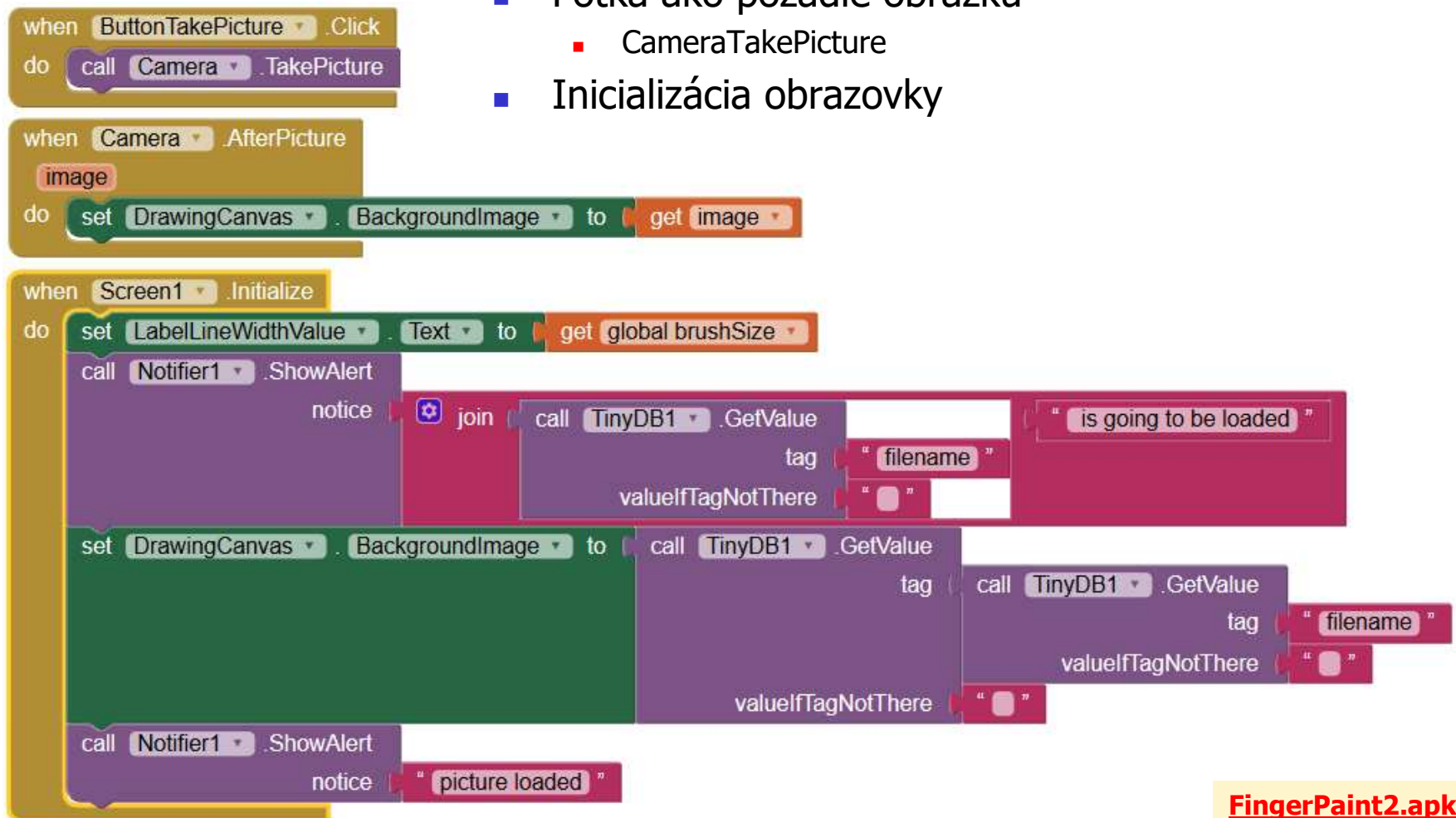
Ak chcete zmazať perzistentné dáta,
Settings/Application/ClearData



FingerPaint2.apk

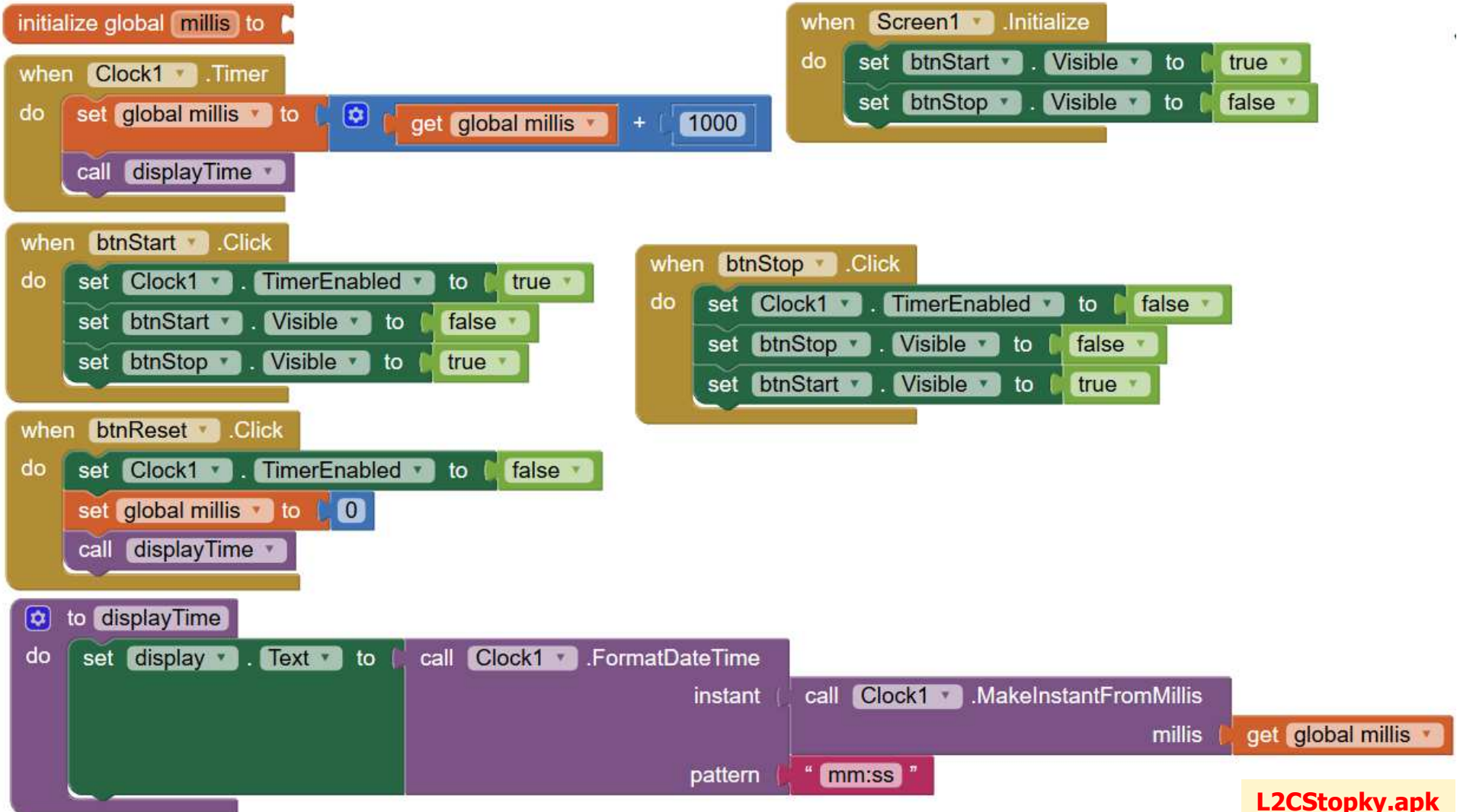
FingerPaint2

- Fotka ako pozadie obrázku
 - CameraTakePicture
- Inicializácia obrazovky



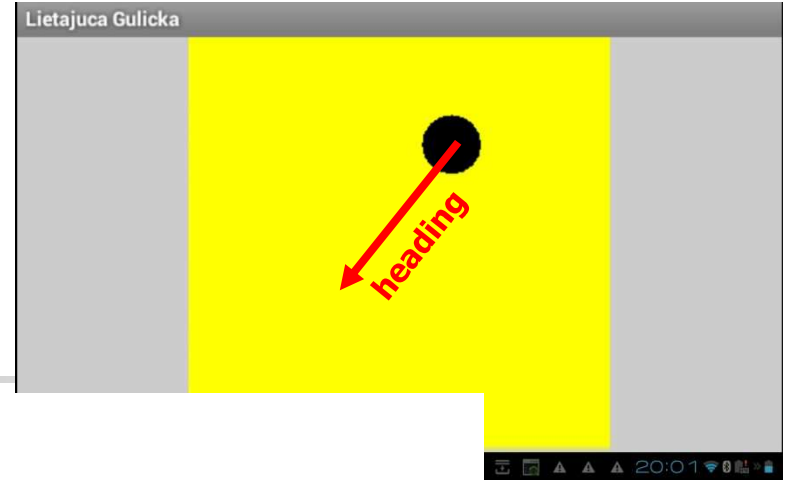
Stopky

(clock – timer)



Dynamic

(random)



```
when Ball1 .EdgeReached
  edge
do
  set Ball1 . Speed to (Ball1 . Speed) * 1.2
  set Ball1 . Radius to (Ball1 . Radius) * 1.2
  set Ball1 . X to (300 * random fraction)
  set Ball1 . Y to (90 * random fraction)

when Clock1 .Timer
do
  set Ball1 . Heading to (360 * random fraction) - 180

when Canvas1 .Touched
  x y touchedAnySprite
do
  call Ball1 .MoveTo
    x (get x)
    y (get y)
```

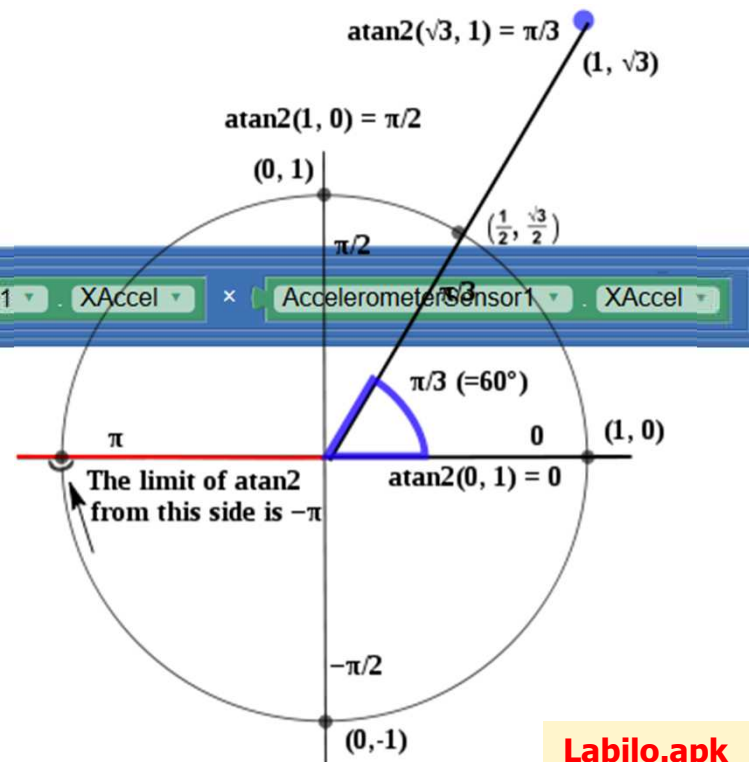
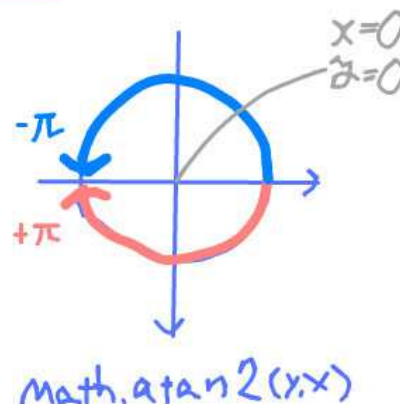
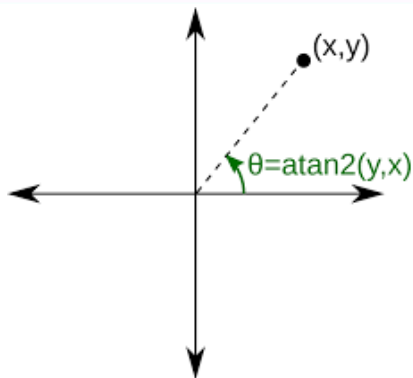
A screenshot of the Scratch IDE. On the left, the "Project" pane shows "Screen1" containing "Canvas1", "Ball1", and "Clock1". On the right, the "Properties" pane for "Clock1" is shown, with "TimerAlwaysFires" and "TimerEnabled" checked, and "TimerInterval" set to 100.

Labilo (atan2)

when AccelerometerSensor1 .AccelerationChanged

do

- set X .Text to round get xAccel
- set Y .Text to round get yAccel
- set Z .Text to round get zAccel
- set Uhol .Text to round atan2
 - y neg get xAccel
 - x get yAccel
- set Ball1 .Heading to round atan2
 - y neg get xAccel
 - x get yAccel
- set Ball1 .Speed to square root AccelerometerSensor1 .XAccel × AccelerometerSensor1 .XAccel



Senzory

(accel, barcode, gyro, NFC, GPS, kompas, pedo, proximity)

Telekom SK 100 % 18

Senzory

Vyzvánací tón

Pedo: 0

Proxi: 5

STLMIČ

Bar Code: Hint for TextBox1 Scan Barcode

Lati: Hint for TextBox1 Longi: Hint for TextBox2

AccelX: 0.26815 AccelY: -0.15323 AccelZ: 9.81623

GyroX: -0.77 GyroY: 0 GyroZ: 0.42

```
when Pedometer1 .StoppedMoving
do set Pedo . Text to Pedometer1 . SimpleSteps
```

```
when BarcodeScanner1 .AfterScan
result
do set BarCode . Text to get result
```

```
when ProximitySensor1 .ProximityChanged
distance
do set Proxi . Text to get distance
```

```
when LocationSensor1 .LocationChanged
latitude longitude altitude speed
do set Lati . Text to get latitude
set Longi . Text to get longitude
```

```
when AccelerometerSensor1 .AccelerationChanged
xAccel yAccel zAccel
do set AccelX . Text to get xAccel
set AccelY . Text to get yAccel
set AccelZ . Text to get zAccel
```

```
when PedoResetBtn .Click
do call Pedometer1 .Reset
call Pedometer1 .Start
```

```
when ScanBtn .Click
do call BarcodeScanner1 .DoScan
```

```
when Clock1 .Timer
do Timer has gone off. Text to Pedometer1 . SimpleSteps
```

```
when GyroscopeSensor1 .GyroscopeChanged
xAngularVelocity yAngularVelocity zAngularVelocity timestamp
do set GyroX . Text to get xAngularVelocity
set GyroY . Text to get yAngularVelocity
set GyroZ . Text to get zAngularVelocity
```

```
when OrientationSensor1 .OrientationChanged
azimuth pitch roll
do set Azim . Text to get azimuth
set Pitch . Text to get pitch
set Roll . Text to get roll
```


Zoznam

(zoznam)

when Screen1.Initialize

do call paint

initialize global zoznam to

- make a list
 - "11-2-pokemon-png.png"
 - "12-2-pokemon-free-download-png.png"
 - "1-2-pokemon-download-png.png"
 - "13-2-pokemon-png-image.png"
 - "3-2-pokemon-png-file.png"
 - "4-2-pokemon-transparent.png"
 - "5-2-pokemon-high-quality-png.png"
 - "7-2-pokemon-png-picture.png"
 - "8-2-pokemon-free-png-image.png"
 - "9-2-pokemon-png-clipart.png"

initialize global index to 0

when NextBtn.Click

do

- set global index to $\text{get global index} + 1$
- call paint

when PrevBtn.Click

do

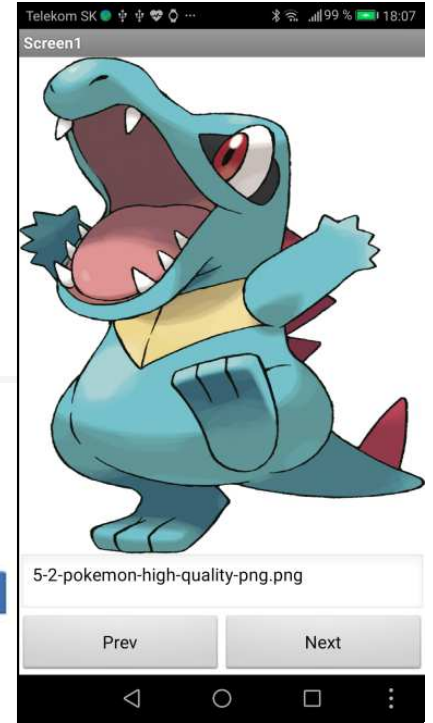
- set global index to $\text{get global index} - 1$

Show Warnings

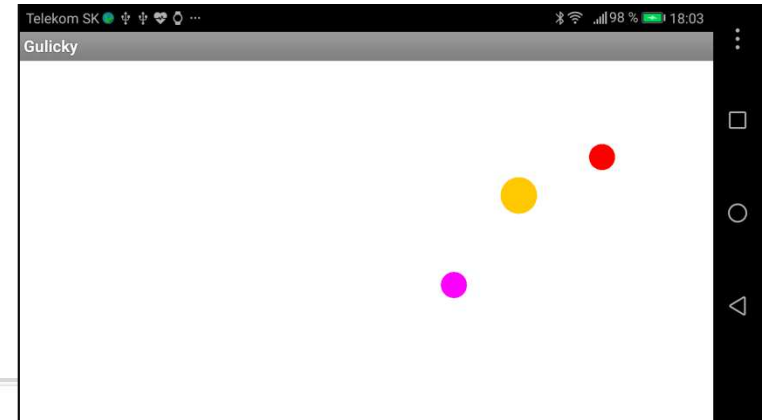
to paint

do

- set global index to $\text{modulo of } \text{get global index} + 10$
- initialize local subor to
 - select list item list get global zoznam index $\text{get global index} + 1$
- in
 - set Canvas1.BackgroundImage to get subor
 - set TextBox1.Text to get subor



Zoznam objektov (for each)



foreach object in a list

```
to init
do
  add items to list list
  item Ball1
  item Ball2
  item Ball3

  for each item in list
  do
    set Ball. X
    of component get item
    to random integer from 1 to 400

    set Ball. Y
    of component get item
    to random integer from 1 to 400

    set Ball. Speed
    of component get item
    to random integer from 3 to 15

    set Ball. Radius
    of component get item
    to random integer from 5 to 15

    set Ball. PaintColor
    of component get item
    to pick a random item list make a list
    red green purple yellow

    set Ball. Heading
    of component get item
    to Ball. Heading of component get item + random integer from -30 to 30
```

TextToSpeech

- Nainštaluj eSpeak
- <https://play.google.com/store/apps/details?id=com.googlecode.eyesfree.espeak&hl=sk>

```
when HearBtn .Click
do call SpeechRecognizer1 .GetText

when SpeechRecognizer1 .AfterGettingText
result partial
do set TextBox1 .Text to SpeechRecognizer1 .Result

when SayBtn .Click
do
  set TextToSpeech1 .Language to LanguageList .Selection
  set TextToSpeech1 .Country to CountryList .Selection
  call TextToSpeech1 .Speak
  message TextBox1 .Text
```

```
when LanguageList .AfterPicking
do
  if LanguageList .Selection = "ENG"
  then set CountryList .Elements to make a list "GBR" "US"

  if LanguageList .Selection = "CES"
  then set CountryList .Elements to make a list "CZE"

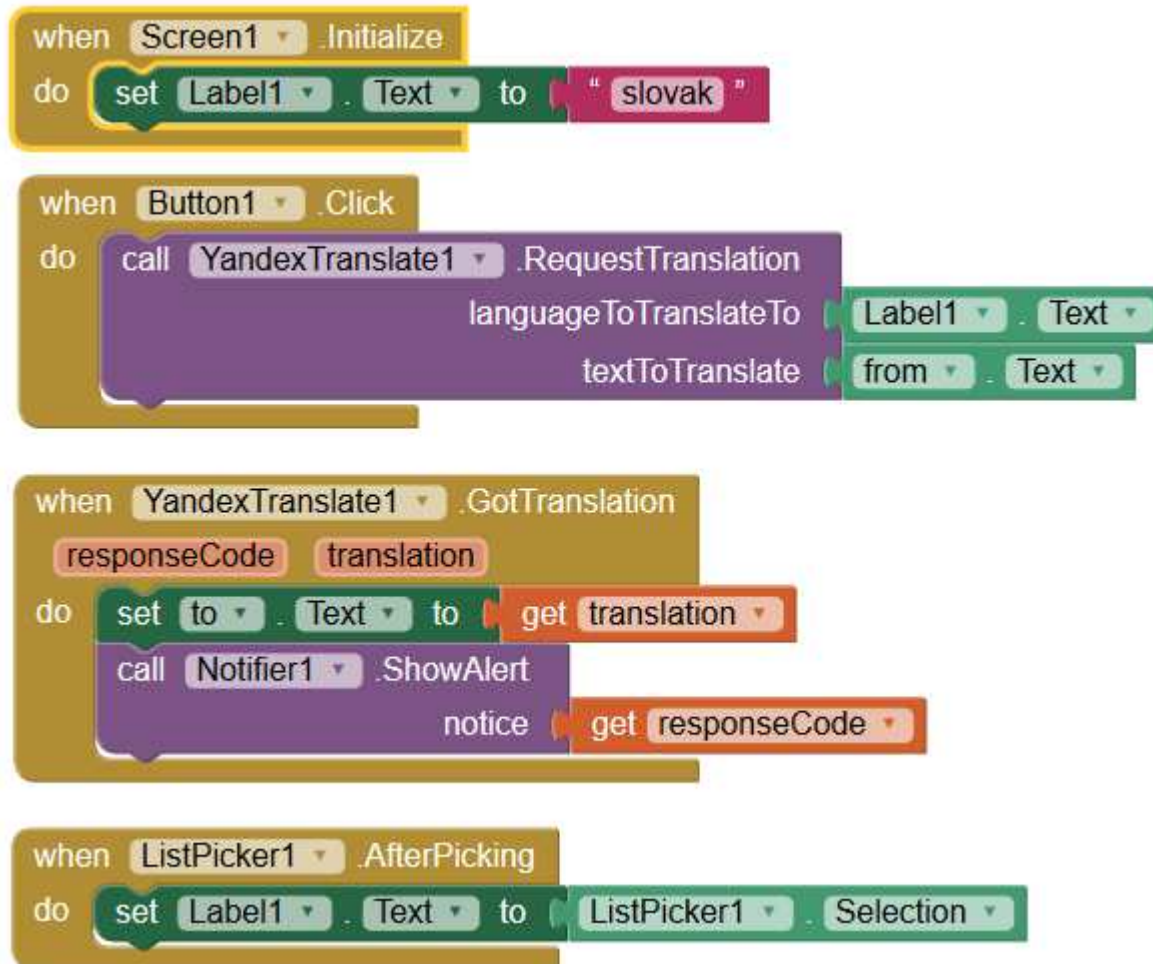
  if LanguageList .Selection = "SPA"
  then set CountryList .Elements to make a list "ESP" "USA"

  if LanguageList .Selection = "DEU"
  then set CountryList .Elements to make a list "AUT" "BEL" "CHE" "LIE" "LUX" "DEU"
```

[Text2Speech.apk](#)

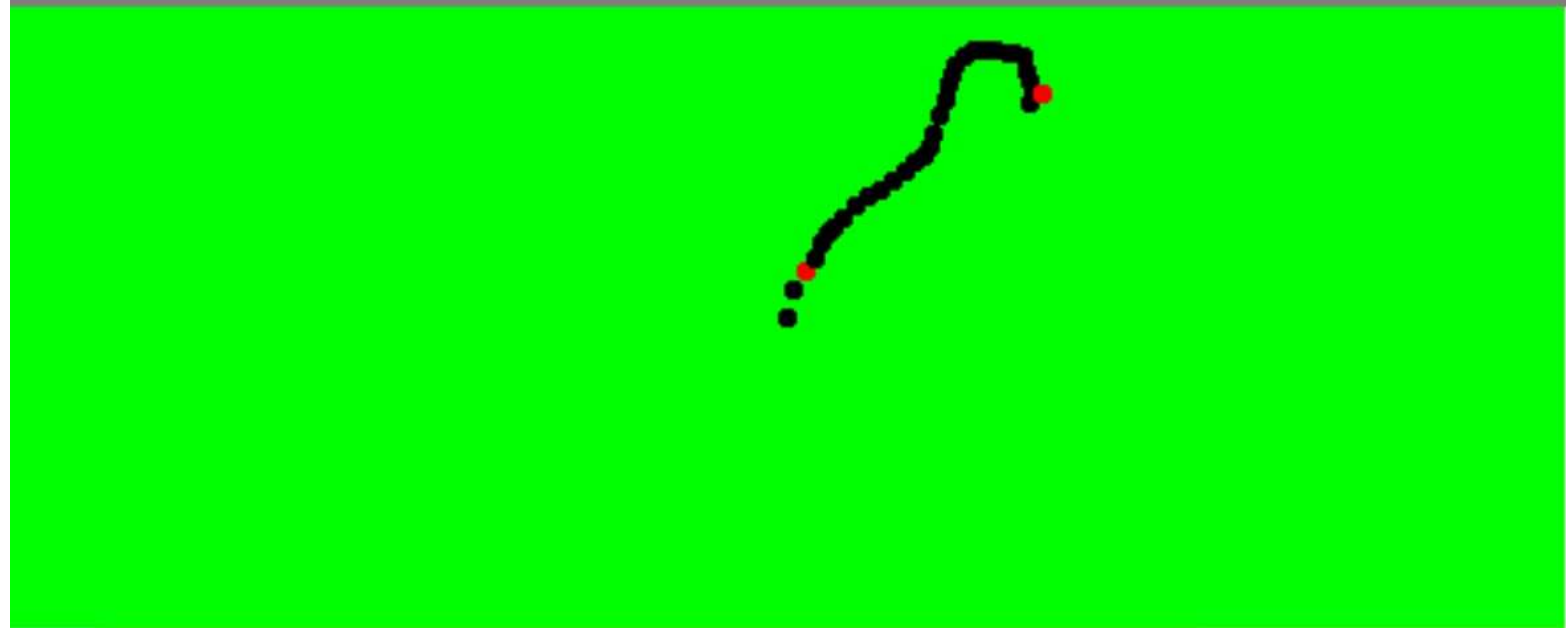


Translator



GPS Art

GPSArt



[x,y] | 82.37588, long.: -71.55161, accur.: 32.0, prov.: gps, adr: Jeséniova 5A 83101 N

Lat 48.16678

Long 17.10979

GPS

Exit

1000



15:26



GPS Art



```
when Screen1.Initialize
do
  set LocationSensor1.ProviderName to "gps"
  set LblConnected.Text to "Initialized"
  set LocationSensor1.Interval to 1000

initialize global initLat to 0
initialize global initLong to 0

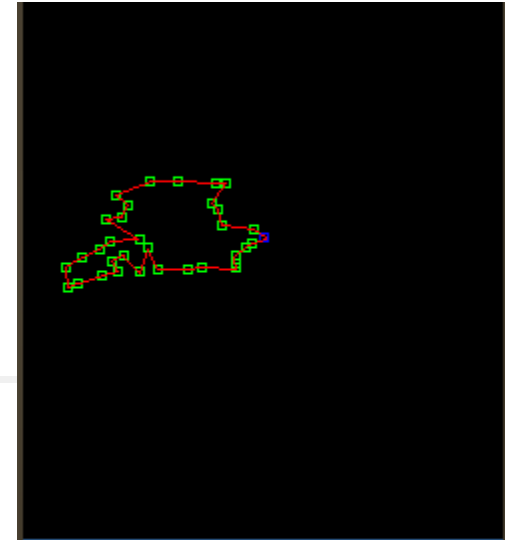
when BtnExit.Click
do
  close screen

when Clock1.Timer
do
  set LocationSensor1.ProviderName to "gps"
  set Canvas1.PaintColor to red
  call ShowPostion
    lat LocationSensor1.Latitude
    long LocationSensor1.Longitude
```

```
when LocationSensor1.LocationChanged
  latitude longitude altitude speed
do
  set Canvas1.PaintColor to black
  call ShowPostion
    lat get latitude
    long get longitude

when BtnGPS.Click
do
  set Canvas1.PaintColor to blue
  call ShowPostion
    lat LocationSensor1.Latitude
    long LocationSensor1.Longitude
```

GPS Art



```
to ShowPosition lat long
do
  set TxtLong . Text to get lat
  set TxtLat . Text to get long
  set LblConnected . Text to LocationSensor1 . TimeInterval
  if
    get global initLong × get global initLat = 0
  then
    set global initLat to get lat
    set global initLong to get long
  set TxtLat . Text to get lat
  set TxtLong . Text to get long
  call Canvas1 . DrawCircle
    centerX
    centerY
    radius
    fill
```

centerX: $(\text{get global initLat} - \text{get lat}) \times 50000 + 250$

centerY: $(\text{get global initLong} - \text{get long}) \times 50000 + 100$

radius: 3

fill: true

GPS Art

