

4.5. Kockový poker

Kľúčové slová

náhodné čísla, sprajty, animácia, ťahanie, pravidlá hry, komunikácia, bluetooth

Čo sa naučíme a čo si precvičíme

- použiť generátor pseudonáhodných čísiel na generovanie náhodných javov v aplikácii,
- animovať a gestami ovládať grafické objekty – sprajty,
- vyhodnocovať stav hry podľa pravidiel,
- komunikovať s iným zariadením prostredníctvom technológie bluetooth.

Príprava na výučbu

Pomôcky: Pracovný list pmz_4_5_Kockovy poker_PL.xlsx na experimentovanie s vyhodnocovaním hodov podľa pravidiel, hracie kocky (odporúčané, nie nutné), základ aplikácie kockovy_poker_v1.aia na preskúmanie a dopracovanie, projekt bluetooth_komunikacia.aia na preskúmanie.

Čo zaujímavé môžeme zistiť?

Kockový poker je spoločenská hra. Podobne ako hry s kartami je nenáročná na prípravu, na hranie nám stačí 6 kociek a niečo na zaznamenávanie výsledkov hry. Hra je založená na náhode (hádže sa kockami), ale hráč v nej robí aj strategické rozhodnutia, či a ako pokračovať v hre, ktorými významne ovplyvňuje priebeh hry.

Existujú rôzne variácie pravidiel kockového pokra. Vo Wikipédii nájdeme takéto:

Ľubovoľný počet hráčov hrá so šiestimi kockami. Hráč začína hru hodom všetkých kociek. Hod sa vyhodnocuje takto:

1. Ak padla trojica, násobí sa počet bodov na kocke stovkou, výnimkou sú tri jednotky, sú za 1000.
2. Každá ďalšia rovnaká kocka pri počte viac ako 3 v jednom hode body zdvojnásobuje.
3. Každá jednotka v počte menej ako 3 je za 100 bodov a každá päťka je za 50 bodov.
4. Postupka, teda čísla od 1 do 6, je za 2000 bodov.

Ak hráč v hode získal aspoň 50 bodov, so zvyšnými kockami môže hádzať ďalej. Ak pri nasledujúcom hode nič nezíska, stráca dovtedy nahraté body v danej hre, ak body získa, môže hádzať ďalej alebo si môže zapísať dosiahnutý výsledok. Ak všetkých šesť kociek získalo body, hráč môže znovu hádzať všetkými kockami, ale s rizikom, že príde o dosiahnuté body.

Hra končí v kole, v ktorom jeden z hráčov dosiahne dopredu dohodnutý súčet bodov (napríklad 5000, 10000 alebo 20000 bodov). Ak jeden z hráčov dosiahol dohodnutý počet bodov, zostávajúci hráči dokončia kolo a vyhráva ten, ktorý získal najviac bodov.

Úloha 1

Zoznámte sa s pravidlami kockového pokra. V elektronickom pracovnom liste pmz_4_5_Kockovy poker_PL.xlsx experimentujte so simuláciou hádzania kociek a sledujte

vyhodnocovanie hodu. Zahrajte si kockový poker v skupine s ozajstnými kockami a vyskúšajte si stratégiu hry s viacerými hodnotami.

hod kockami	1	3	4	4	4	6
	1	2	3	4	5	6
počet výskytov	1	0	1	3	0	1
bodovanie	100	0	0	400	0	0
						500

Obr. 4.5.1 Príklad vyhodnotenia simulovaného hodu 6 kockami

Akú zaujímavú aplikáciu môžeme vytvoriť?

Hoci rekvizity na hranie kockového pokra nie sú objemné, predsa len nie je bežné, že človek nosí so sebou 6 hracích kociek. Problém pri hraní môže byť tiež s hádzaním kociek, ak nemáme na to vhodné prostredie. Mobilný telefón s aplikáciou môže simulovať náhodné hody kockami, asistovať pri vyhodnocovaní hry a zaznamenávaní výsledkov, a tak nahradiť všetky rekvizity potrebné k hre a uľahčovať administrovanie výsledkov.

Naprogramujeme aplikáciu, ktorá bude:

- simulovať hod kockami,
- umožňovať rozhodovanie hráča o pokračovaní resp. ukončení hry,
- bodovo vyhodnocovať hru hráča: body za aktuálny hod, body za všetky hody v rámci jedného ťahu, celkový súčet bodov,
- zaznamenávať celkové skóre hráča.

Takáto aplikácia realizuje hru jedného hráča. Viacerí hráči môžu hrať každý so svojím telefónom a výsledky si evidovať každý vo svojom zariadení. Elegantnejšie riešenie kockového pokra ako spoločenskej hry je, ak mobilné aplikácie hráčov budú medzi sebou komunikovať a aplikácie budú riadiť priebeh hry a zaznamenávanie výsledkov skupiny hráčov. Naša aplikácia bude určená pre dvoch hráčov a bude:

- evidovať hráča na ťahu,
- blokovať hru pre hráča, ktorý nie je na ťahu,
- evidovať a zobrazovať výsledky oboch hráčov,
- oznamovať ukončenie hry a víťaza.

Ako budeme postupovať pri tvorbe aplikácie?

Najprv vytvoríme aplikáciu pre jedného hráča. Potom pridáme komunikáciu a riadenie hry pre dvoch hráčov.

Úloha 2

Preskúmajte aplikáciu `kockovy_poker_v1.aia`.

Čo obsahuje globálna premenná `kocky` a v ktorom bloku programu sa jej priradujú hodnoty? Na čo sa premenná v programe používa?

Pomoc k riešeniu úlohy

Premenná `kocky` je zoznam šiestich komponentov typu `ImageSprite`, ktoré predstavujú v aplikácii hracie kocky. Zoznam kociek sa vytvorí pri inicializácii obrazovky (udalosť `Screen1.Initialize`). Zoznam sa používa na hromadné spracovanie komponentov rovnakého typu v cykle pomocou blokov zo skupiny *Any Component*.

Čo obsahuje globálna premenná `hodnotyKociek` a v ktorom bloku programu sa jej nastavujú hodnoty?

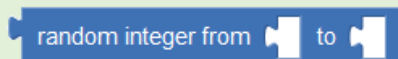
Pomoc k riešeniu úlohy

Premenná `hodnotyKociek` je zoznam šiestich náhodných čísiel z intervalu 1 až 6, ktoré predstavujú hodnoty kociek po hode. Nastavujú sa v procedúre `hodKockami`.

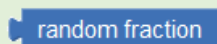
Vysvetlíme si

Hod kockou je náhodný jav, ktorý sa nedá predpovedať, možno iba určiť jeho pravdepodobnosť. Ak predpokladáme dokonalú kocku, tak pravdepodobnosť padnutia ľubovoľného čísla na kocke je rovnaká pre všetky čísla 1 až 6, a to $1/6$. Algoritmicke sa dajú generovať postupnosti čísiel s vlastnosťami, ktoré zodpovedajú teoretickým pravdepodobnostiam. Takéto algoritmy sa nazývajú generátory pseudonáhodných čísiel.

V App Inventore sa generujú pseudonáhodné celé čísla zo zadaného intervalu pomocou funkcie



a pseudonáhodné reálne čísla z intervalu $<0,1)$ pomocou funkcie



Na čo slúži premenná `fazyAnimacie`?

Pomoc k riešeniu úlohy

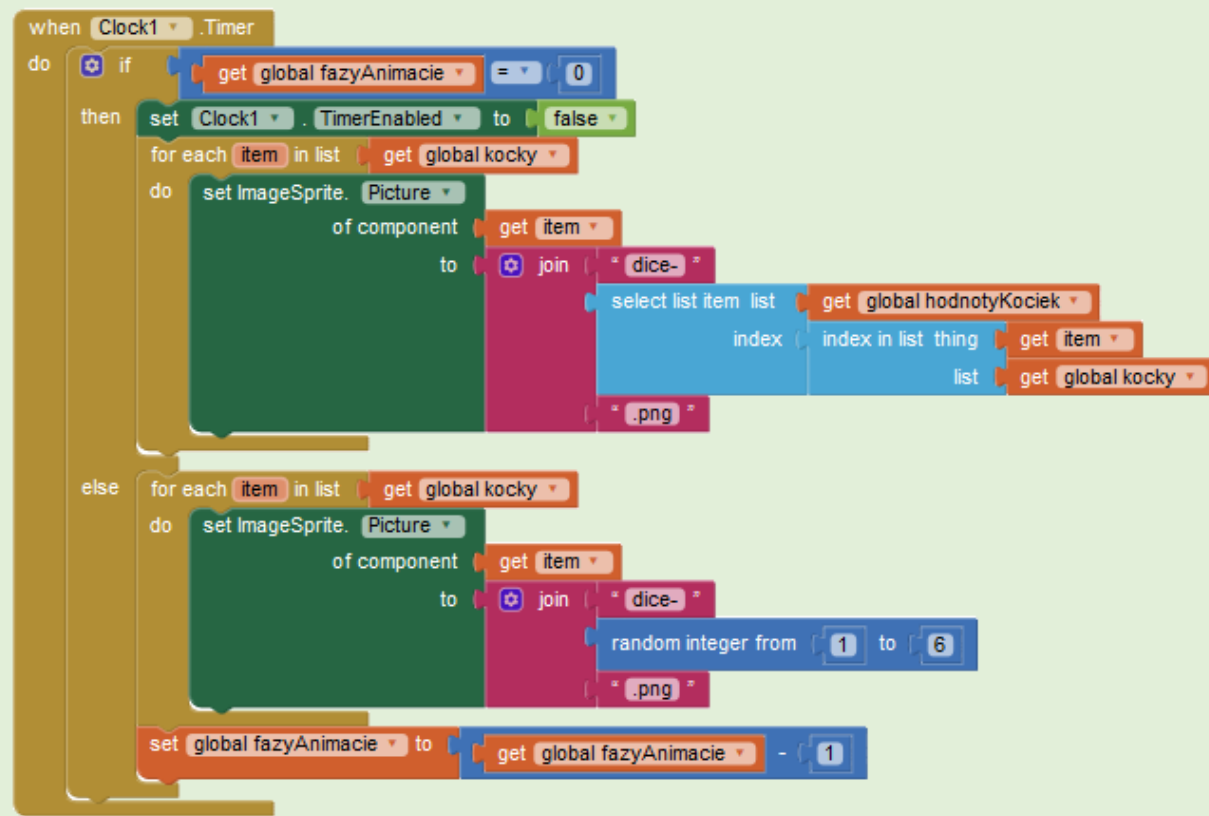
Projekt obsahuje šesť obrázkov kocky pre šesť možných hodnôt, ktoré môžu pri hode padnúť. Pri hode kockou sa vygeneruje náhodná hodnota ako výsledok hodu a kocke sa nastaví príslušný obrázok. Aby vznikol dojem pohybu pri hode, vystrieda sa pred nastavením konečného obrázku kocky niekoľko ďalších obrázkov podľa náhodne vygenerovaných hodnôt. Premenná `fazyAnimacie` obsahuje počet fáz, ktorými má prejsť obrázok kocky pri animácii hodu, kým sa ustáli na hodnote vygenerovanej v zozname `hodnotyKociek`.

Vysvetlíme si

Animácia je striedanie statických obrázkov tak, aby sa vytvoril dojem pohybu.

Striedanie obrázkov pri animácii robíme v projekte pomocou komponentu `Clock1` a jeho udalosti `Timer`. Pri hode kockami sa aktivuje časovač komponentu `Clock1` nastavením vlastnosti `TimerEnabled` na `true`. Komponent začne generovať v pravidelných

intervaloch udalosti `Timer`. Pri každom tiknutí časovača sa náhodne nastaví obrázok hodených kociek a zníži sa hodnota premennej `fazyAnimacie` o 1. Po vynulovaní nastaveného počtu fáz animácie sa obrázok kocky nastaví podľa konečnej hodnoty uloženej v premennej `hodnotyKociek`, časovač komponentu `Clock1` sa deaktivuje nastavením vlastnosti `TimerEnabled` na `false` a animácia sa zastaví. Animuje sa hod všetkých kociek uložených v zozname `kocky` v cykle `for each item in list kocky`.



Čo sa deje v aplikácii pri ťahaní a pri pustení kociek?

Pomoc k riešeniu úlohy

Pri ťahaní (udalosť `Dragged` komponentu `ImageSprite`) sa kocka presúva po hracom priestore (komponent `Canvas`). Pri pustení sa vráti do pôvodnej pozície v rade kociek, ak bola pustená v ľavej polovici, inak sa zaradí na miesto v rade kociek v pravej polovici hracieho priestoru.

Vysvetlíme si

`ImageSprite` (sprajt) je vizuálny komponent, ktorý môže byť umiestnený v komponente `Canvas`. Môže sa pohybovať a reagovať na dotykové gestá.

Vybrané vlastnosti komponentu:

`Picture` – obrázok, ktorý definuje vzhľad sprajtu,

`Height`, `Width` – rozmery (výška, šírka) sprajtu,

`X`, `Y` – súradnice ľavého horného rohu sprajtu.

Vybrané udalosti:

`Dragged(startX, startY, prevX, prevY, currentX, currentY)` – ťahanie, parametre v obslužnej metóde udalosti udávajú pozíciu začiatku ťahania, predchádzajúcej a aktuálnej pozície sprajtu

`TouchDown(x, y)` – chytenie, parametre udávajú pozíciu, na ktorej bol sprajt uchopený

`TouchUp(x, y)` – pustenie, parametre udávajú pozíciu, na ktorej bol sprajt pustený

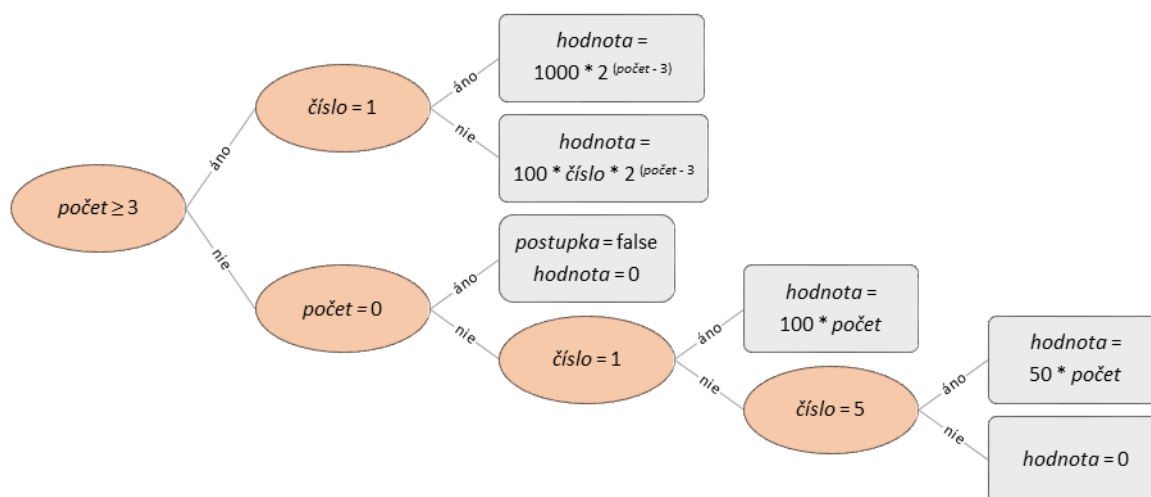
Vybraná metóda:

`MoveTo(x, y)` – premiestni sprajt na dané súradnice

Úloha 3

Naprogramujte funkciu `vyhodnotenie` na bodové vyhodnotenie hodu podľa pravidiel hry. Výsledok vypíšte.

Bodové vyhodnotenie hodu robíme pre každé číslo, ktoré môže na kockách padnúť (1 až 6). Spočítame počet výskytov čísla v hode (hodnoty kociek v aktuálnom hode sú uložené v premennej `hodnotyKociek`). Z počtu výskytov čísla vypočítame bodovú hodnotu podľa rozhodovacieho stromu na obrázku 4.5.2.

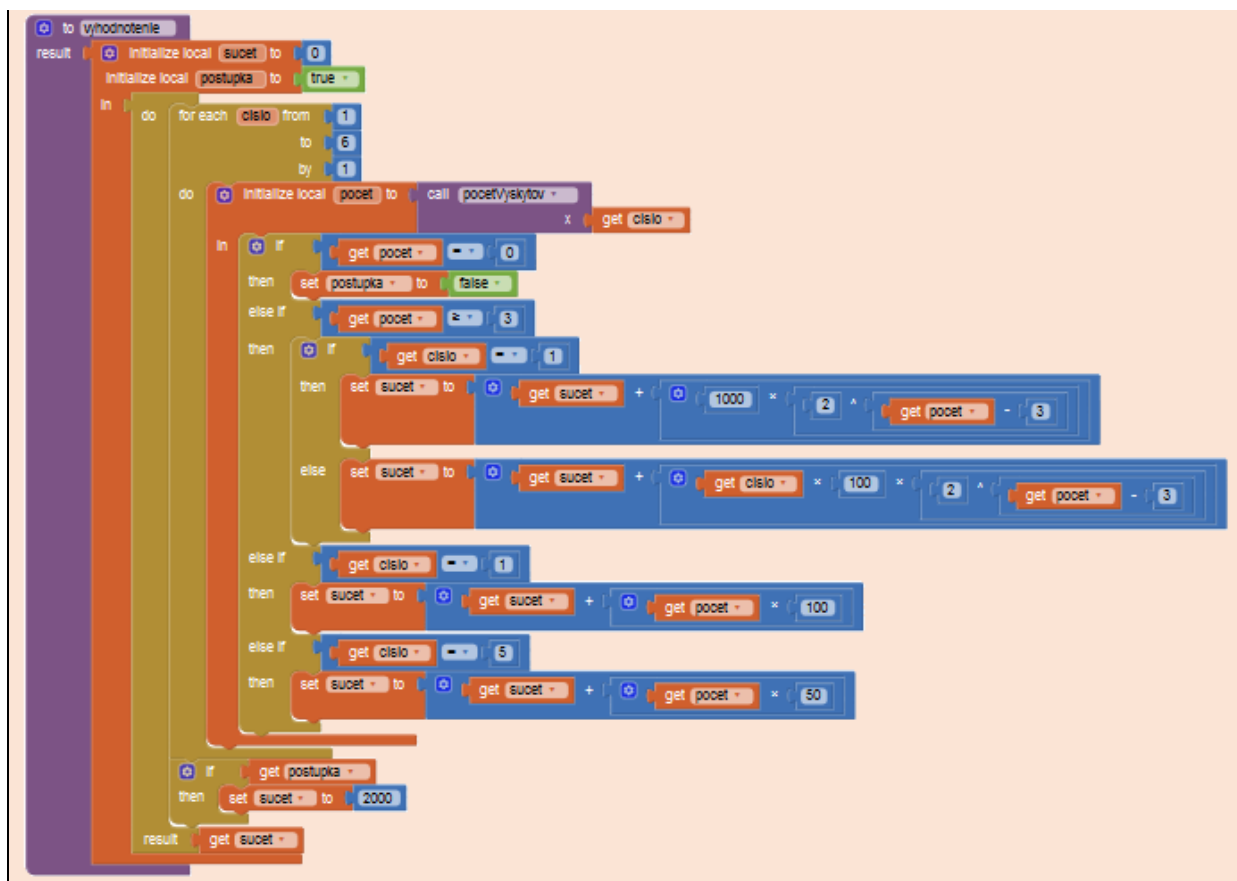


Obr. 4.5.2 Rozhodovací strom

Celkovú bodovú hodnotu hodu získame sčítaním bodových hodnôt pre všetky čísla. Výnimkou je postupka, kedy sa hodnota hodu rovná 2000, a nie súčtu hodnôt pre jednotlivé čísla. Na zistenie výskytu postupky slúži v rozhodovacom strome logická premenná **postupka**, ktorú na začiatku inicializujeme na true a na false prepisujeme, ak sa zistí nulový výskyt nejakého čísla v hode. Príklad vyhodnotenia hodu je na obrázku 4.5.1.

Pomoc k riešeniu úlohy

Funkcia `vyhodnotenie` počíta dosiahnutý počet bodov pri hode kockami podľa rozhodovacieho stromu na obr. 4.5.2. Využíva informáciu o počte výskytov jednotlivých čísiel 1 až 6 v danom hode (uložené v zozname `hodnotyKociek`) zistenú pomocnou funkciou `pocetVyskytov(x)`, kde parameter `x` je číslo, ktorého počet výskytov v hode sa zisťuje.

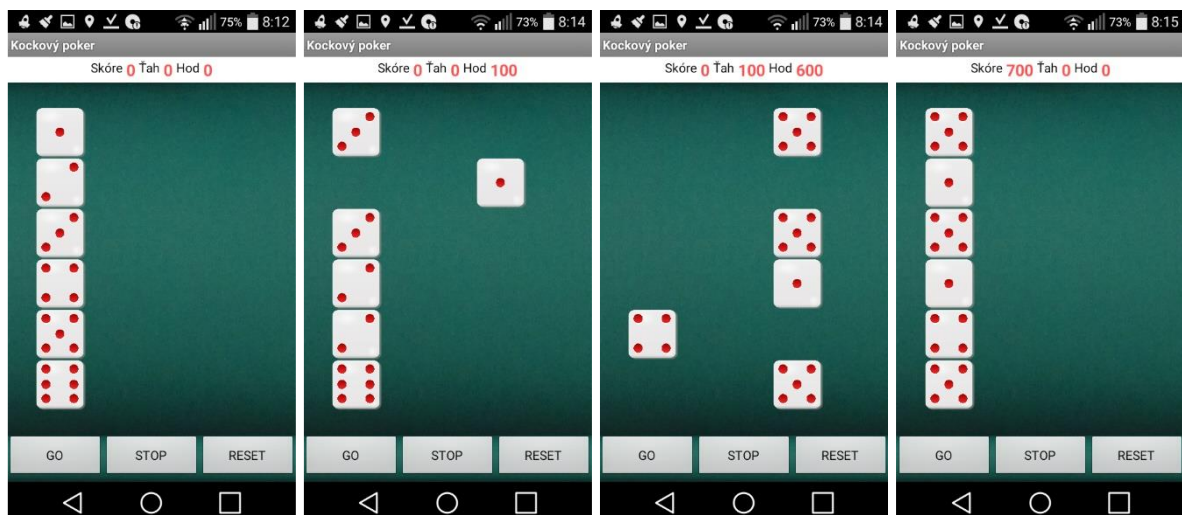


Úloha 4

Upravte funkciu `vyhodnotenie` tak, aby vyhodnocovala len kocky vybrané používateľom na pravú stranu hracej plochy. Zobrazujte zvlášť body za aktuálny hod, priebežný súčet bodov v jednom ťahu a celkové skóre hráča.

Na obrázku 4.5.3 sú snímky obrazovky, ktoré zachytávajú priebeh jednej hry:

1. úvodné rozloženie kociek,
2. po prvom hode šiestimi kockami (stlačenie tlačidla **GO**): napravo kocka vybraná používateľom na bodovanie, naľavo kocky bez bodovej hodnoty, hodnota hodu 100,
3. po druhom hode so zostávajúcimi piatimi kockami (po stlačení tlačidla **GO**): priebežná hodnota hry 100 + hodnota aktuálneho hodu 600
4. po ukončení ťahu (hry) stlačením tlačidla **STOP**: celkové skóre 700



Obr. 4.5.3 Priebeh hry

Pomoc k riešeniu úlohy

Kockám pridáme ďalšie vlastnosti:

bodovaná – je true ak je kocka vybraná hráčom na bodovanie presunutím na pravú stranu obrazovky, inak false; pre 6 kociek sú uložené v zozname `bodovane`

aktívna – je true, ak je kocka v hre a je viditeľná na obrazovke, na false sa zmení po započítaní hodnoty kocky a jej vyradení z ďalšej hry; pre 6 kociek sú uložené v zozname `aktivne`

Úloha 5

Preskúmajte projekt `komunikacia_bluetooth.aia` – funkčnosť aplikácie a programový kód.

Aplikácia z projektu umožňuje komunikáciu medzi dvomi zariadeniami pomocou technológie bluetooth.

Vysvetlíme si

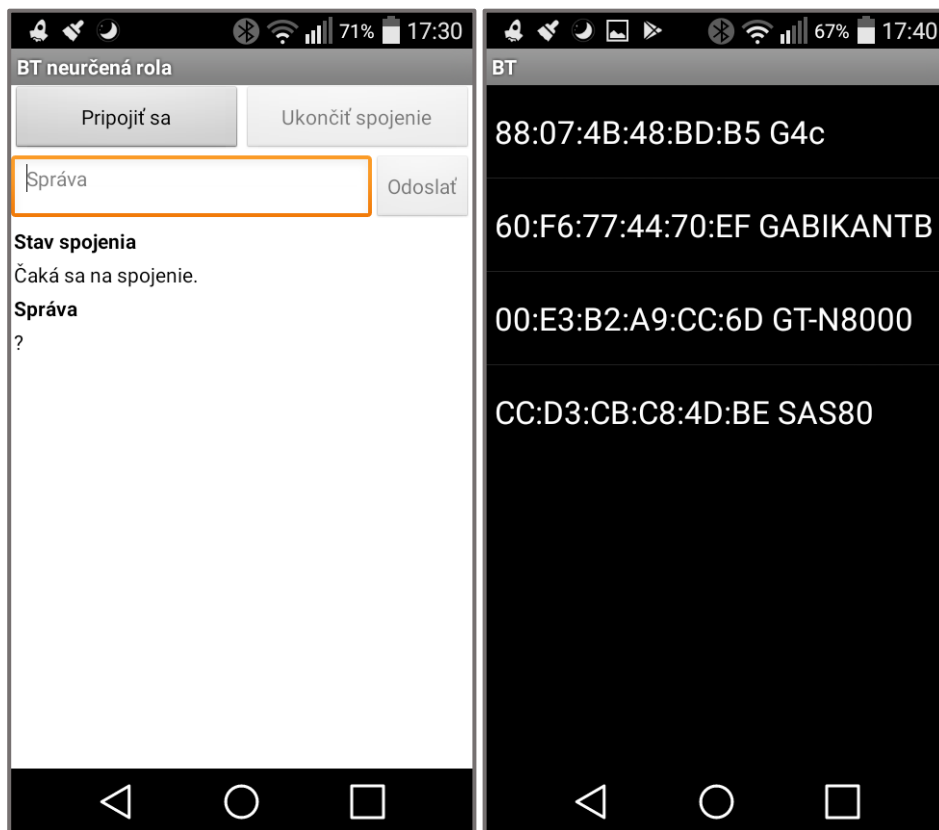
Bluetooth je technológia pre bezdrôtovú komunikáciu medzi dvomi alebo viac digitálnymi zariadeniami. Pri spojení komunikujúcich zariadení sa využíva architektúra klient-server. Zariadenie, ktoré začína komunikáciu, je klient, zariadenie, ktoré prijíma žiadosť o komunikáciu, je server.

V App Inventore slúžia na vytvorenie spojenia bluetooth komponenty `BluetoothClient` a `BluetoothServer` zo skupiny komponentov *Connectivity*.

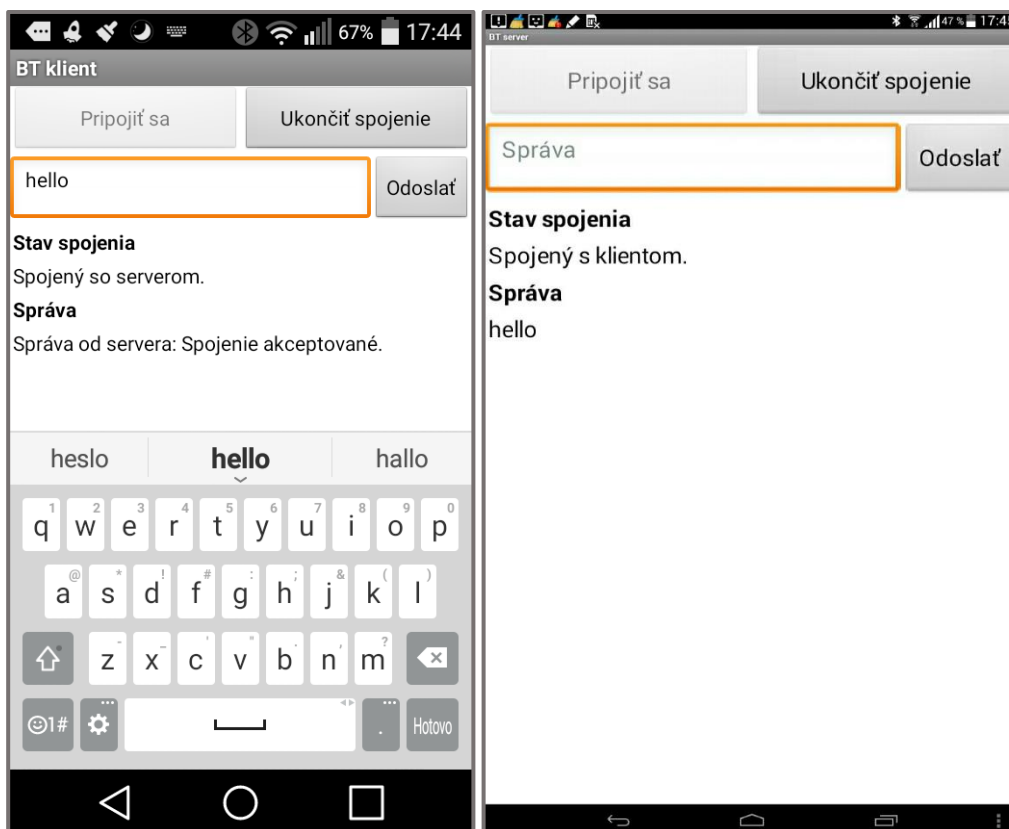
Na otestovanie funkčnosti aplikácie budeme potrebovať dve mobilné zariadenia. Príkazom **Build** vytvoríme Inštalačný balík aplikácie (.apk) a do oboch zariadení nainštalujeme a spustíme ukážkovú aplikáciu `komunikacia_bluetooth`. V oboch zariadeniach zapneme bluetooth a zariadenia spárujeme.

Spojenie začína jedno zo zariadení stlačením tlačidla s nápisom **Pripojiť**. Toto zariadenie bude vystupovať v úlohe klienta. Zo zoznamu sa vyberie zariadenie, ku ktorému sa chceme pripojiť.

To bude v úlohe servera. Ak server akceptuje žiadosť klienta, spojenie sa úspešne nadviaže. Funkčnosť spojenia môžeme vyskúšať odoslaním správ z jedného do druhého zariadenia.

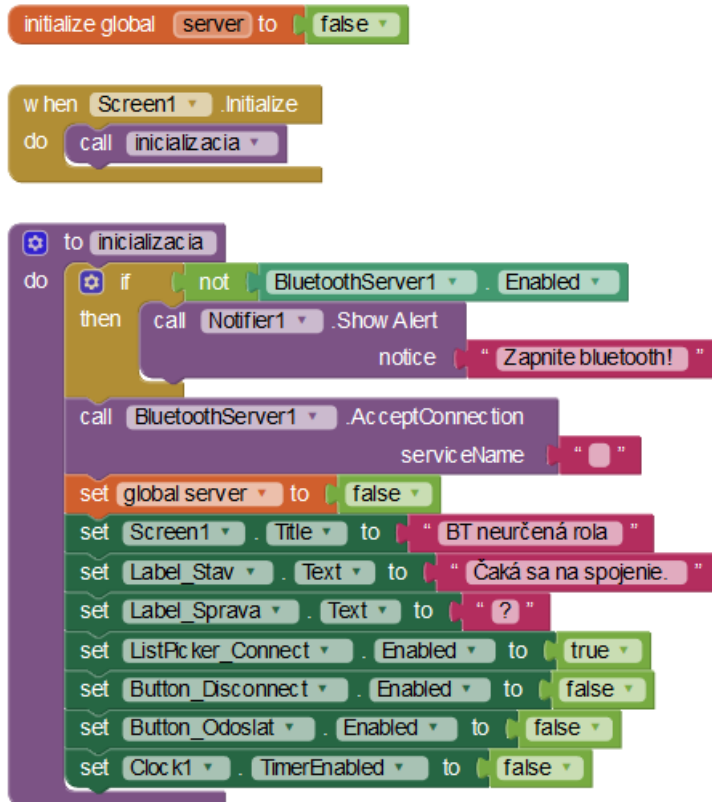


Obr. 4.5.4 Inicializácia spojenia (tlačidlo Pripojiť), výber zariadenia na spojenie zo zoznamu



Obr. 4.5.5 Aplikácia v režime klient a server

Preskúmajme program. Pri inicializácii aplikácia skontroluje, či je bluetooth zapnutý (vlastnosť `BluetoothServer.Enabled`) a nastaví serverovú časť na akceptovanie spojenia metódou `BluetoothServer.AcceptConnection`.



Oba režimy (klient aj server) sú naprogramované v jednej aplikácii. O tom, či bude aplikácia pracovať v režime klient alebo server, rozhodne to, kto z dvojice iniciuje spojenie. Na začiatku je aplikácia v režime server a čaká na žiadosť o spojenie.

Aplikácia bude pracovať v režime *klient*, keď používateľ iniciuje spojenie stlačením komponentu s textom **Pripojiť sa**. Je to komponent typu `ListPicker` – výber zo zoznamu. Reakcie na udalosti komponentu `ListPicker`:

- udalosť `ListPicker.BeforePicking` (pred výberom) – zoznam (`ListPicker.Elements`) sa naplní menami a adresami dostupných a spárovaných bluetooth zariadení (`BluetoothClient.AddressesAndNames`).
- udalosť `ListPicker.AfterPicking` (po výbere)
 - ak medzitým serverová časť aplikácie akceptovala žiadosť o spojenie (`BluetoothServer.IsAccepting`), akceptácia sa zruší `BluetoothServer.StopAccepting`
 - ak sa spojenie s vybraným zariadením podarilo (`BluetoothClient.Connect (ListPicker.Selection)`), odošle sa správa serveru o úspešnom pripojení klienta a aplikácia sa nastaví do režimu klient, premenná `server` bude `false`
 - inak sa aplikácia znovu inicializuje

```

when ListPicker_Connect .BeforePicking
do
  if not BluetoothClient1 .Enabled
  then
    call Notifier1 .ShowAlert
    notice "Zapnite bluetooth!"
  set ListPicker_Connect .Elements to BluetoothClient1 .AddressesAndNames

```

```

when ListPicker_Connect .AfterPicking
do
  set Label_Stav .Text to join "pokús o pripojenie k "
  ListPicker_Connect .Selection
  if BluetoothServer1 .IsAccepting
  then
    call BluetoothServer1 .StopAccepting
  if
    call BluetoothClient1 .Connect
    address ListPicker_Connect .Selection
  then
    set global server to false
    set Screen1 .Title to "BT klient"
    set ListPicker_Connect .Enabled to false
    set Button_Disconnect .Enabled to true
    set Button_Odoslat .Enabled to true
    set Clock1 .TimerEnabled to true
    call odoslat
    sprava "Správa od klienta: pripojený."
    set Label_Stav .Text to "Spojený so serverom."
  else
    call inicializacia

```

Aplikácia pracujúca v režime *server* neiniciuje spojenie, ale prijíma žiadosť o spojenie od druhého zariadenia, vtedy sa generuje udalosť:

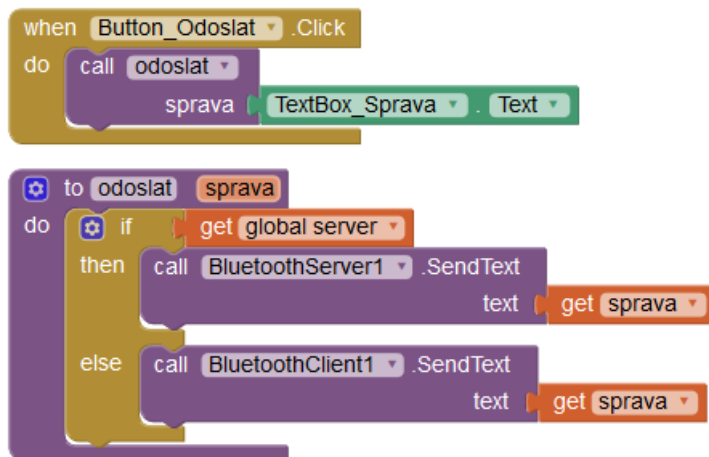
- `BluetoothServer.ConnectionAccepted` – aplikácia sa nastaví do režimu *server* (premenná `server` sa nastaví na `true`) a odošle sa správa klientovi o úspešnom pripojení.

```

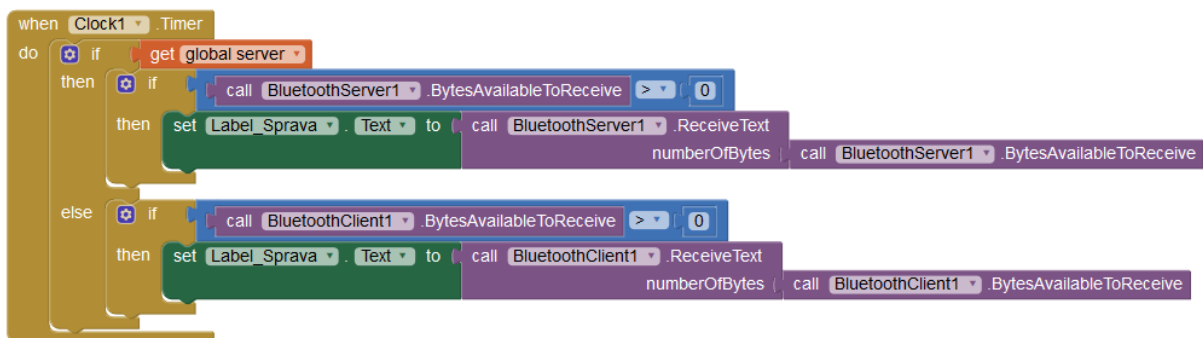
when BluetoothServer1 .ConnectionAccepted
do
  set global server to true
  set Screen1 .Title to "BT server"
  set ListPicker_Connect .Enabled to false
  set Button_Disconnect .Enabled to true
  set Button_Odoslat .Enabled to true
  set Clock1 .TimerEnabled to true
  set Label_Stav .Text to "Spojený s klientom."
  call odoslat
  sprava "Správa od servera: Spojenie akceptované."

```

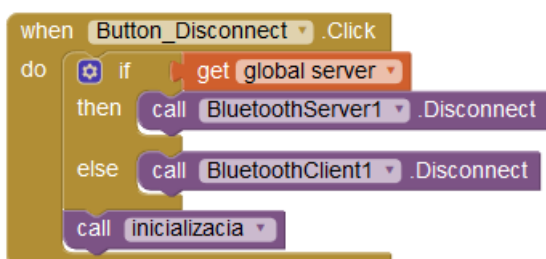
Po úspešnom spojení si zariadenia môžu vymieňať správy. Na odoslanie správy sa použije metóda `SendText` komponentu `BluetoothClient` alebo komponentu `BluetoothServer` podľa toho, v akom režime aplikácia pracuje.



Na prijímanie správ je použitý komponent `Clock` ako časovač. Časovač generuje pravidelné udalosti (`Timer`), pri ktorých sa zisťuje, či bola prijatá správa. Funkcia `BytesAvailableToReceive` komponentu `BluetoothClient` alebo komponentu `BluetoothServer` podľa toho, v akom režime aplikácia pracuje, zistí počet prijatých bajtov. Ak je väčší ako 0, funkcia `ReceiveText` vráti prijatý text.



Spojenie sa ukončí stlačením tlačidla s nápisom **Ukončiť spojenie**. V reakcii na udalosť `Click` komponentu `Button` sa zavolá metóda `Disconnect` komponentu `BluetoothClient` alebo komponentu `BluetoothServer` podľa toho, v akom režime aplikácia pracuje.



Úloha 6

Pridajte do projektu Kockový poker komunikáciu medzi dvomi hráčmi. Podľa vzoru ukážkovej aplikácie z úlohy 5 nadviažte spojenie dvoch zariadení. Doprogramujte funkčnosť aplikácie:

- po ukončení ťahu
 - aplikácia odošle pripojenému zariadeniu informácie: body za ťah a celkové skóre,
 - aplikácia obmedzí hráčovi ďalší ťah dovtedy, kým nedostane správu o ukončení ťahu súpera,
- po obdržaní správy o ukončení ťahu súpera:
 - aplikácia zobrazí prijaté informácie o skóre súpera a vyhodnotí priebežný stav hry,
 - ak nie je koniec hry, aplikácia sprístupní hráčovi nový ťah,
 - ak je koniec hry, vyhodnotí víťaza.

Ako vylepšiť či rozšíriť našu aplikáciu?

Hráči kockového pokra používajúci našu aplikáciu hrajú každý so svojím mobilným zariadením a odosielajú si prostredníctvom bluetooth komunikácie informácie o dosiahnutých bodoch vo svojom ťahu. Rozšírením aplikácie by mohlo byť „zdieľanie obrazovky“ súpera: Hráči by si posielali všetky informácie o hádzaní a výbere kociek, na základe ktorých by sa na obrazovke mohol zobrazovať celý priebeh hry súpera (animácia hodov, výber kociek), nielen výsledné bodové hodnotenie.

Čo sme sa naučili

- ✓ Rozumieť architektúre klient-server pri komunikácii prostredníctvom technológie bluetooth.
- ✓ Použiť komponenty `BluetoothClient` a `BluetoothServer` na vytvorenie klientskej a serverovej aplikácie na komunikáciu dvoch mobilných zariadení.
- ✓ Vytvoriť univerzálnu aplikáciu s klientskou a serverovou časťou komunikácie bluetooth.

Odporúčaný priebeh výučby		
Činnosť učiteľa	Činnosť žiaka	Poznámky
<i>1. hodina – Úvod, úlohy 1, 2</i>		
Učiteľ oboznámi žiakov s témou projektu – spoločenská hra kockový poker.	Žiaci sa zapájajú do diskusie o hrách v mobile.	Je dôležité vzbudiť záujem žiakov o tému projektu, preto diskusii o téme projektu venujeme dostatočný čas.
Učiteľ vysvetlí pravidlá kockového pokra.	Žiaci sa zoznamujú s pravidlami hry a bodovaním (úloha 1) simulovaním hodov v pracovnom liste, prípadne aj hraním hry s reálnymi kockami.	Simulácia hodov a ich vyhodnocovanie demonštruje pravidlá bodovania a naznačuje algoritmus vyhodnocovania hodu.
Učiteľ rozdelí žiakov do dvojíc. Zadá úlohu 2 a pomáha žiakom pri skúmaní základu projektu.	Žiaci odteraz pracujú vo dvojiciach. Skúmajú základ projektu, ktorý majú doprogramovať.	Práca so začatým projektom ušetrí čas pri vývoji aplikácie. Keďže výsledkom projektu bude hra pre dvoch hráčov,

		žiaci pracujú vo dvojiciach kvôli testovaniu hry.
2. hodina – úlohy 3 a 4		
Učiteľ pomáha pri problémoch s riešením úloh.	Žiaci vo dvojiciach programujú funkcie na vyhodnocovanie hodu kockami. Používajú jedno mobilné zariadenie a živé ladenie.	Výsledkom práce bude funkčná hra zatiaľ pre jednu osobu. Hru môžu žiaci otestovať nainštalovaním do dvoch zariadení, striedavo hádžu kockami každý vo svojom mobile.
3. a 4. hodina – úlohy 5, 6		
Učiteľ vysvetľuje princíp komunikácie bluetooth a demonštruje jej programovanie v App Inventore na ukážkovej aplikácii bluetooth_komunikacia.aia.	Žiaci si nainštalujú ukážkovú aplikáciu do mobilov. Vo dvojiciach testujú jej funkčnosť. Potom skúmajú programový kód aplikácie s pomocou učiteľa.	Porozumenie riešenia v ukážkovej aplikácii je potrebné pre ďalšiu prácu na projekte.
Učiteľ pomáha s realizáciou bluetooth komunikácie v projekte Kockový poker.	Žiaci aplikujú riešenie z ukážkovej aplikácie vo svojom projekte. Hru testujú na dvoch zariadeniach.	Pri testovaní hry sa nedá používať živé ladenie, treba vždy vytvoriť inštalačný súbor aplikácie (Build) a inštalovať ho do dvoch mobilov.
5. hodina – prezentovanie, hodnotenie projektov		
Učiteľ organizuje prezentáciu projektov. Ohodnotí riešenia žiakov.	Žiaci prezentujú svoje projekty učiteľovi a spolužiakom, testujú si navzájom aplikácie, vyjadrujú názor na riešenia spolužiakov.	Záverečné hodnotenie v určenom čase je vyvrcholením práce na projekte. Hodnotenie musí byť štruktúrované podľa zadania projektu.

Bibliografia

Michaličková, V. (2016). *Programovanie mobilných aplikácií v prostredí MIT App Inventor 2*. Nitra: Univerzita Konštatína Filozofa v Nitre.

MIT. (2018). Retrieved from: <http://ai2.appinventor.mit.edu/>

Pura Vida Apps (2018). *A Simple Bluetooth Chat with App Inventor 2*. Retrieved from: <https://puravidaapps.com/btchat.php>

Wolber, D., Abelson, H., Spertus, E., & Liz, L. (2014). *App Inventor 2 - Create Your Own Android Apps*. O'Reilly.

Wikipedia. (2018). *Kockový poker*. Dostupné na internete: https://sk.wikipedia.org/wiki/Kockový_poker

Register pojmov

A

animácia, 3

B

bluetooth, 7
BluetoothClient, 7
BluetoothServer, 7

G

generátor pseudonáhodných čísiel, 3

R

random, 3
rozhodovací strom, 5

S

sprajt, 4