



# App Inventor – programovanie mobilných aplikácií

Učebný materiál pre učiteľov pre ZŠ

# Pre učiteľa



Tento materiál slúži ako základný podklad pre prácu učiteľa s rozhraním App Inventor pre účely programovania mobilných aplikácií na ZŠ. Okrem návodov na vytvorenie základných aplikácií sa tu nachádza aj rámec, podľa ktorého si je dobré učivo rozdeliť do jednotlivých časových úsekov.

Predpokladáme, že každý učiteľ prešiel školením, čo je eduScrum a ako ho použiť vo vyučovacom procese. V každom šprinte žiaci môžu pracovať na ľubovoľných aplikáciách, mali by však na konci mať spoznámkované dané témy, ktoré sú predmetom šprintu. Či už v rámci aplikácií, na ktorých pracovali, alebo ako teóriu rozhrania App Inventor. Výber je na nich.

## Úvod

Vysvetlenie metódy eduScrum a spôsob práce so žiakmi. Rozhranie App Inventor.

## Prvý šprint

Premenné

Čísla v programe

Textové reťazce v programe

Príprava poznámok – tu pracujeme s textovým editorom. Ideálne online Google Dokumenty.

Príprava demo prezentácie

Demo prezentácia predošlého šprintu.

## Druhý šprint

Cykly 1

Cykly 2

Podmienky 1

Príprava poznámok – tu pracujeme s textovým editorom. Ideálne online Google Dokumenty.

Príprava demo prezentácie

Demo prezentácia predošlého šprintu.

## **Tretí šprint**

Podmienky 2

Funkcie 1

Funkcie 2

Príprava poznámok – tu pracujeme s textovým editorom. Ideálne online Google Dokumenty.

Príprava demo prezentácie

Demo prezentácia predošlého šprintu.

## **Štvrtý šprint**

Práca s obrázkami

Práca so zvukmi

Ikonka aplikácie

Príprava poznámok – tu pracujeme s textovým editorom. Ideálne online Google Dokumenty.

Príprava demo prezentácie

Demo prezentácia predošlého šprintu.

## **Piaty šprint**

Práca na vlastnej aplikácii alebo hre 1

Práca na vlastnej aplikácii alebo hre 2

Práca na vlastnej aplikácii alebo hre 3

Príprava poznámok/dokumentácie – tu pracujeme s textovým editorom. Ideálne online Google Dokumenty.

Príprava demo prezentácie

Demo prezentácia predošlého šprintu.

## **Šiesty šprint**

Práca na vlastnej aplikácii alebo hre 4

Práca na vlastnej aplikácii alebo hre 5

Práca na vlastnej aplikácii alebo hre 6

Príprava poznámok/dokumentácie – tu pracujeme s textovým editorom. Ideálne online Google Dokumenty.

Príprava demo prezentácie

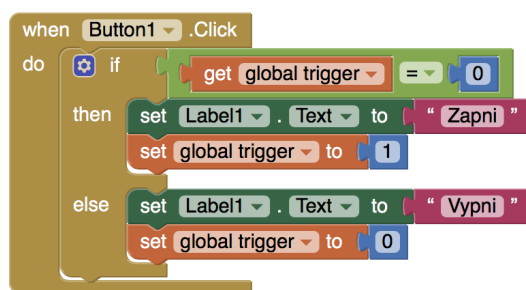
Demo prezentácia predošlého šprintu.

# AppInventor – ako začať

MIT App Inventor ([appinventor.mit.edu](http://appinventor.mit.edu)) je nástroj, pomocou ktorého si môže ako dieťa, tak i stará mama urobiť Android appku.

## Čo je AppInventor

Je to nástroj, ktorý vznikol na MIT ([web.mit.edu/](http://web.mit.edu/)). Jeho úlohou je pomôcť pri začiatkoch programovania mobilných aplikácií. Konkrétne pre Android. Jeho výhodou je, že začiatočník nemusí poznať programovací jazyk v jeho čistej forme, pretože celé programovanie prebieha vizuálne.



**Obrázok 1.** Vizuálne programovanie vyzerá nejak takto.

S trochou znalosti z angličtiny sa však dá vyskladať pomerne zložitá aplikácia.

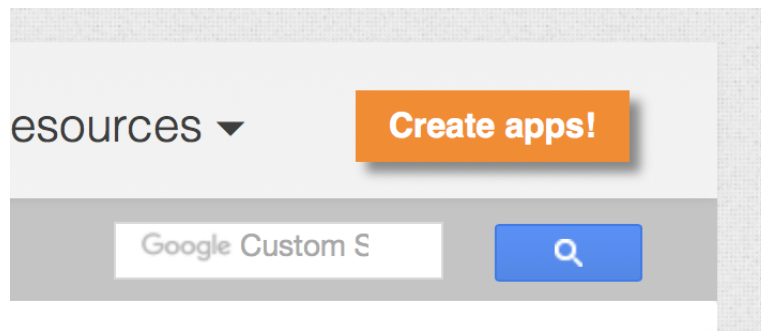
## Pre koho je AppInventor

Pre každého, kto doteraz neprogramoval, trochu sa toho obáva, no zároveň ho to láka. Je určený pre deti v ľubovoľnom veku, pre staré mamy, starých otcov, pre kohokoľvek.

Ideálny je na výučbu programovania mobilných aplikácií na úrovni základných škôl, kde je nutné, aby deti najskôr spoznali princípy algoritmického myslenia na praktickom projekte. Rovnako sa hodí aj na stredné školy, kde študenti nemajú predchádzajúci znalosť s programovacím jazykom. Tu je to vhodné ako vstupná brána do pokročilého programovania.

## Ako urobiť prvú appku

Stačí navštíviť web [appinventor.mit.edu](http://appinventor.mit.edu) a registrovať alebo prihlásiť sa pomocou tlačidla „Create Apps!“.



**Obrázok 2.** Tlačidlo „Create Apps!“

Celý proces registrácie prebieha pomocou Google účtu, ktorý musí mať používateľ vytvorený. Bez neho sa nikam nedostane.

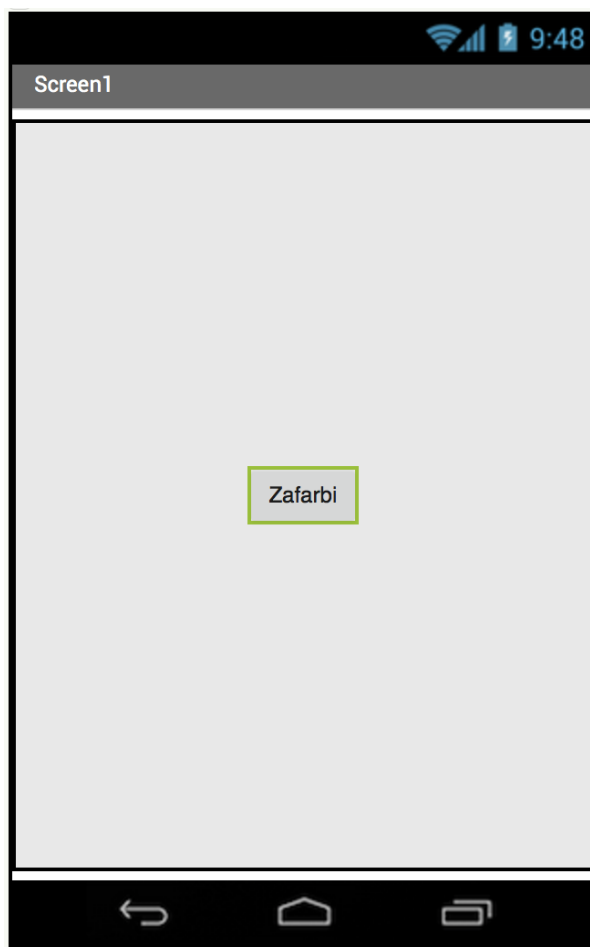
Následne vstúpi dnu, kde sa už dá vytvoriť vlastná appka.

# Ako urobiť prvú appku: zafarbenie pozadia stlačením tlačidla

Vytvorenie prvej aplikácie je jednoduchšie, ako uvariť čaj.

## Farebný prst

To najjednoduchšie, čo môžeme naprogramovať, je pridať jedno tlačidlo do obrazovky a po jeho stlačení zmeniť farbu pozadia.

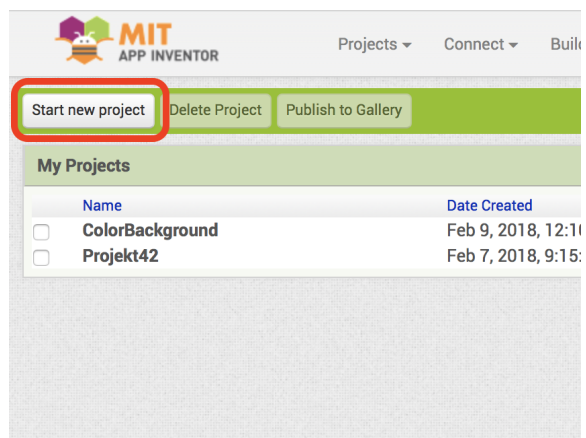


**Obrázok 1.** Jednoduché rozhranie aplikácie.

Ako urobiť toto rozhranie si rozoberieme postupne.

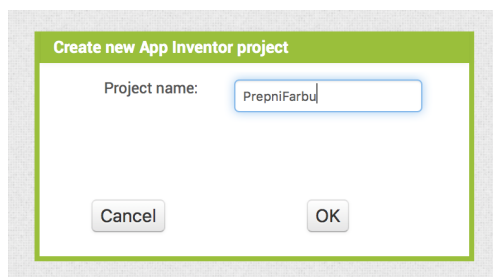
## Krok #0 – Vytvorenie projektu

Po prihlásení sa do AppInventora vytvor nový projekt.



**Obrázok 2.** Vytvorenie nového projektu.

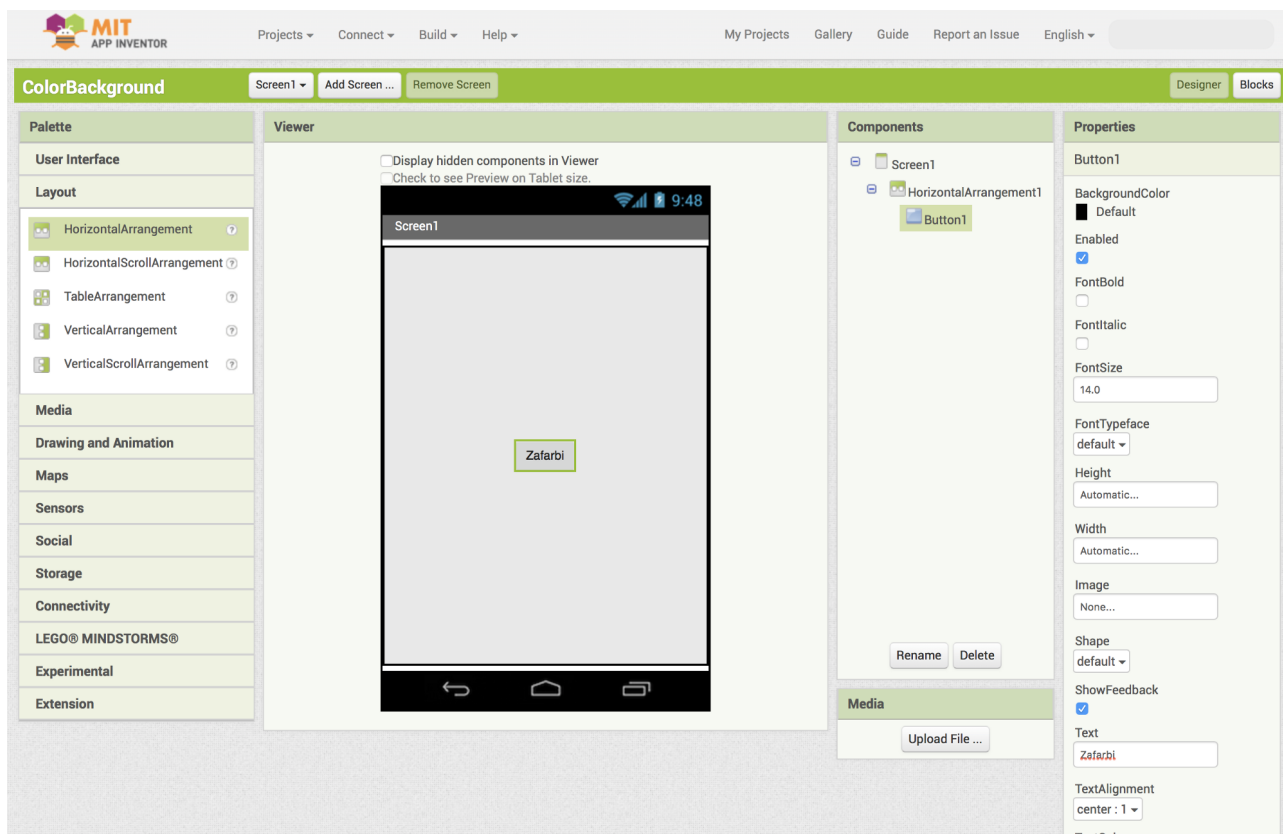
Následne sa objaví okno na zadanie mena projektu bez medzier.



**Obrázok 3.** Pomenovanie projektu.

Klikni na OK.

## Krok #1 – Popis prostredia



Obrázok 4. Prostredie, kde vytvoríme appku.

### Časť Palette

V tejto časti úplne naľavo nájdeme paletu všetkých možných prvkov, ktoré odtiaľto môžeme ťahať do časti **Viewer**.

### Časť Viewer

Tu sa modeluje vizuálna časť aplikácie. Všetky prvky ako tlačidlá, prepínače, ovládače, obrázky, zoznamy a vôbec.

### Časť Components

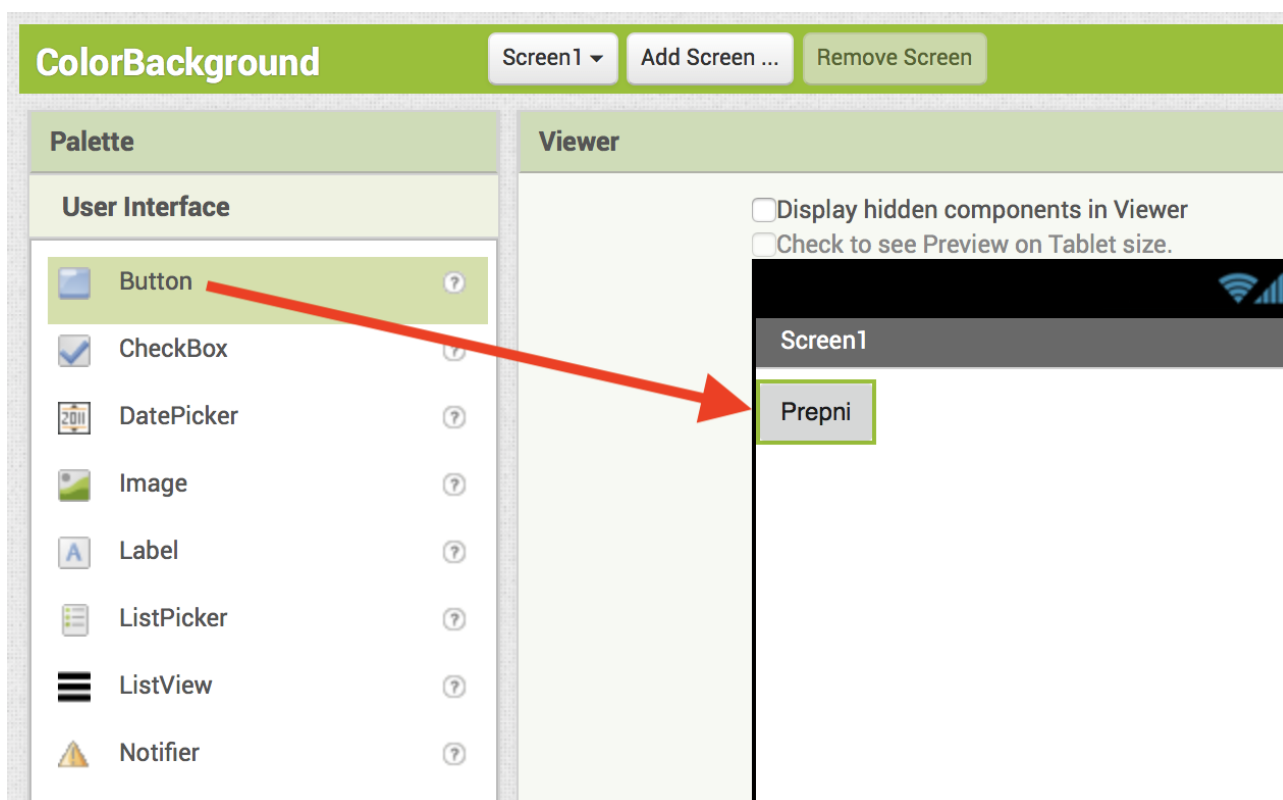
Táto časť zobrazuje štruktúru prvkov, ktoré sme poťahali z časti **Components** do časti **Viewer**. Totiž, keď sa nejaký prvok stratí na obrazovke pod iným, v tejto časti ho určite nájdeme v prehľadnom strome.

## Časť Properties

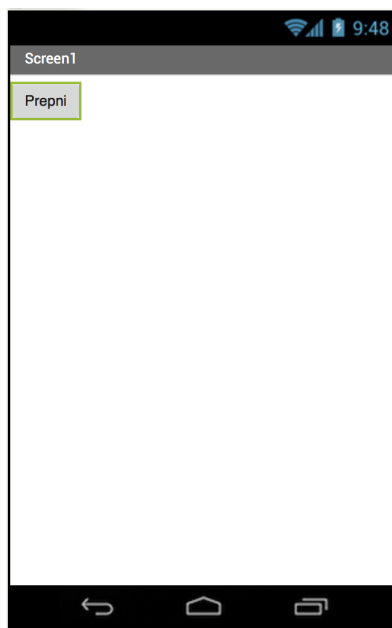
Tu nastavujeme vlastnosti prvku, ktorý sme kliknutím označili. Vlastnosti súvisia s označeným prvkom. Tlačidlo ich bude mať iné, ako obrázok. Podrobne sa k nim dostaneme v konkrétnom prípade.

## Krok #2 – Pridanie tlačidla

Tlačidlo pridáme tak, že v časti **Palette** otvoríme sekciu **User Interface**, v ktorej chytíme tlačidlo Button a ťaháme do obrazovky na požadované miesto.

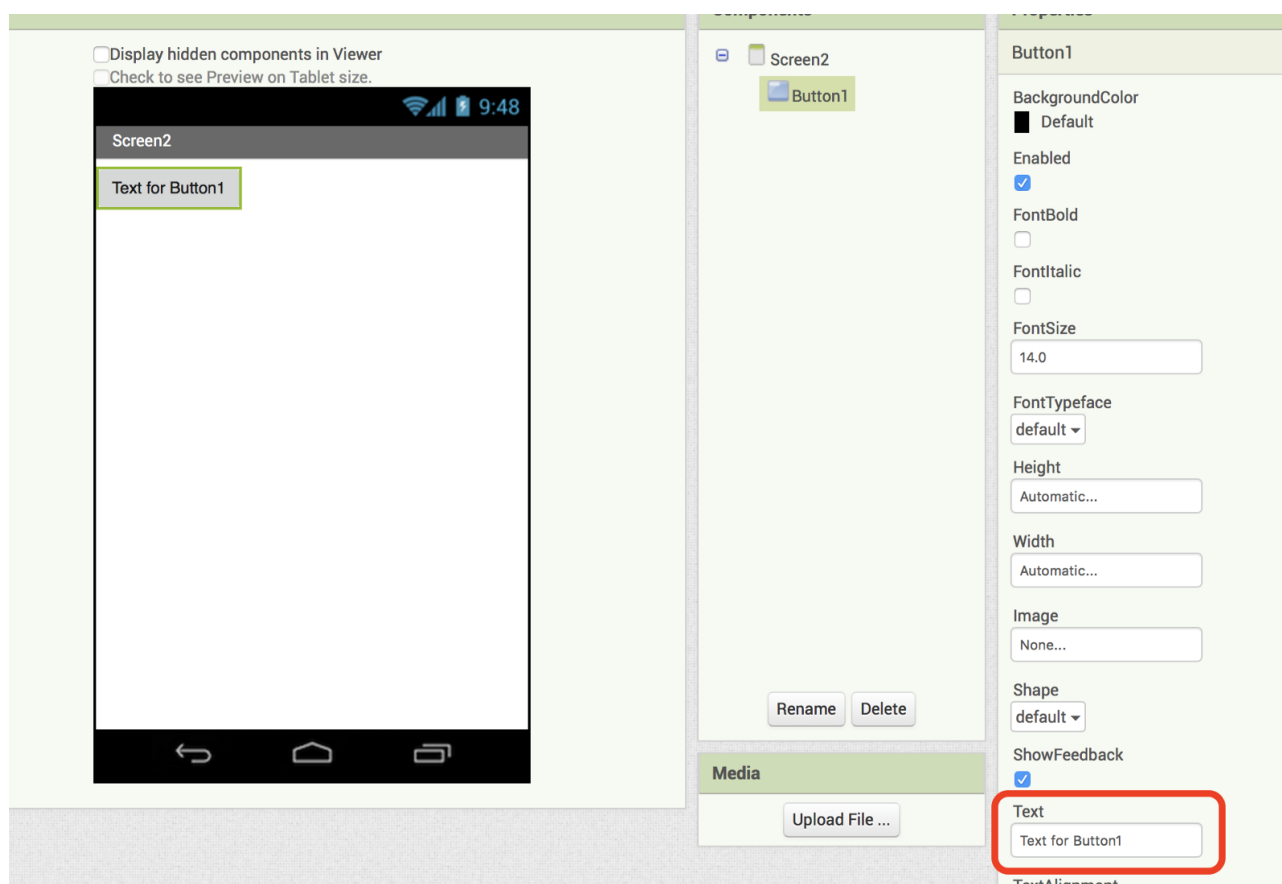


**Obrázok 5.** Potiahnutie tlačidla do obrazovky aplikácie.



**Obrázok 6.** Tlačidlo v obrazovke.

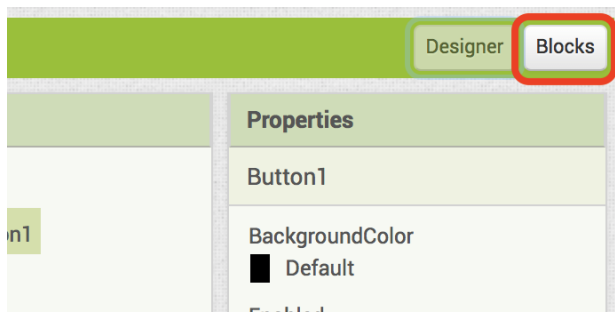
Keď mu chceme pridať vlastný text tak, ako je na obrázkoch vyššie: „Prepni“, tak ho označíme a v časti **Properties** zmeníme jeho vlastnosť **Text** na ľubovľná názov.



**Obrázok 7.** Zmena textu tlačidla.

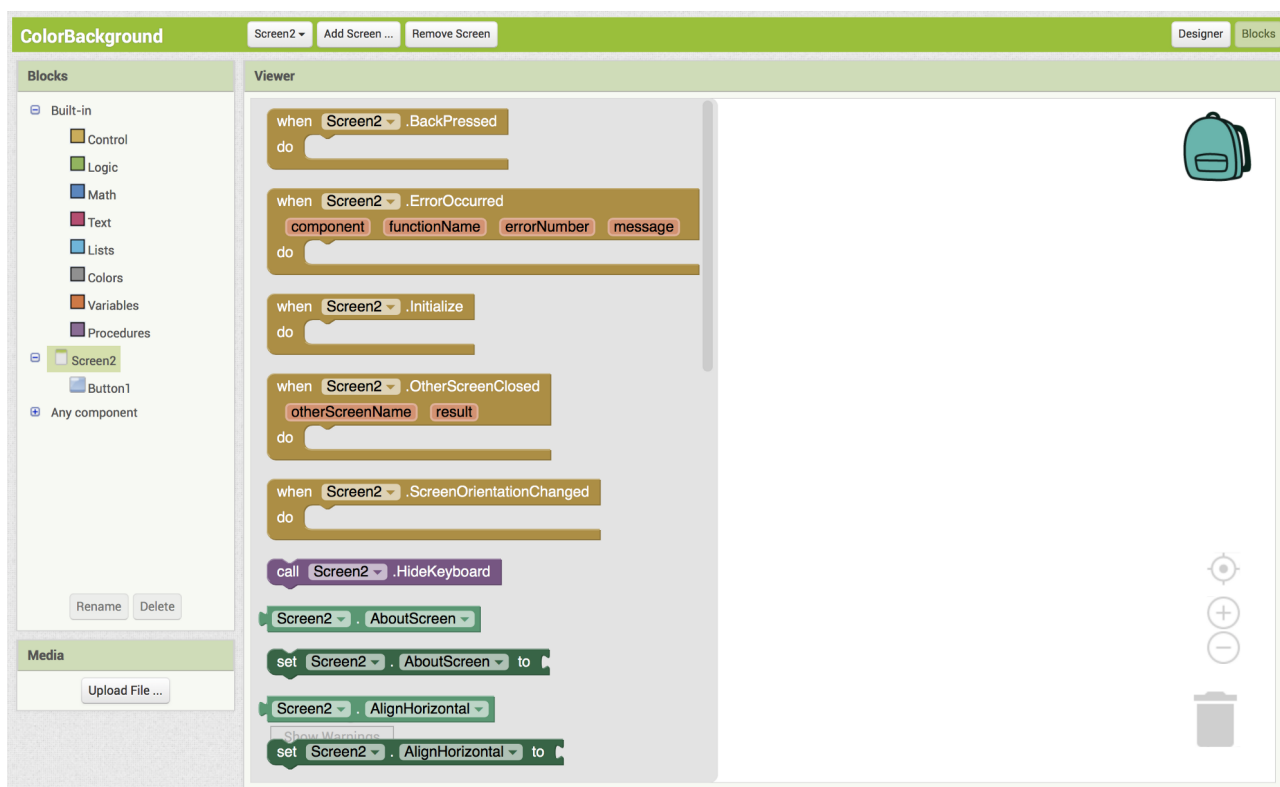
### Krok #3 – Sfunkčnenie tlačidla

Ak chceme, aby tlačidlo niečo urobilo, keď ju spustíme na telefóne alebo tablete, musíme ho naprogramovať. Programovanie prebieha v časti **Blocks**, do ktorej sa musíme prepnúť.



**Obrázok 8.** Prepnutie do časti Blocks, kde programujeme.

Obrazovka sa následne rozdelí na dve základné časti: **Blocks** a **Viewer**.

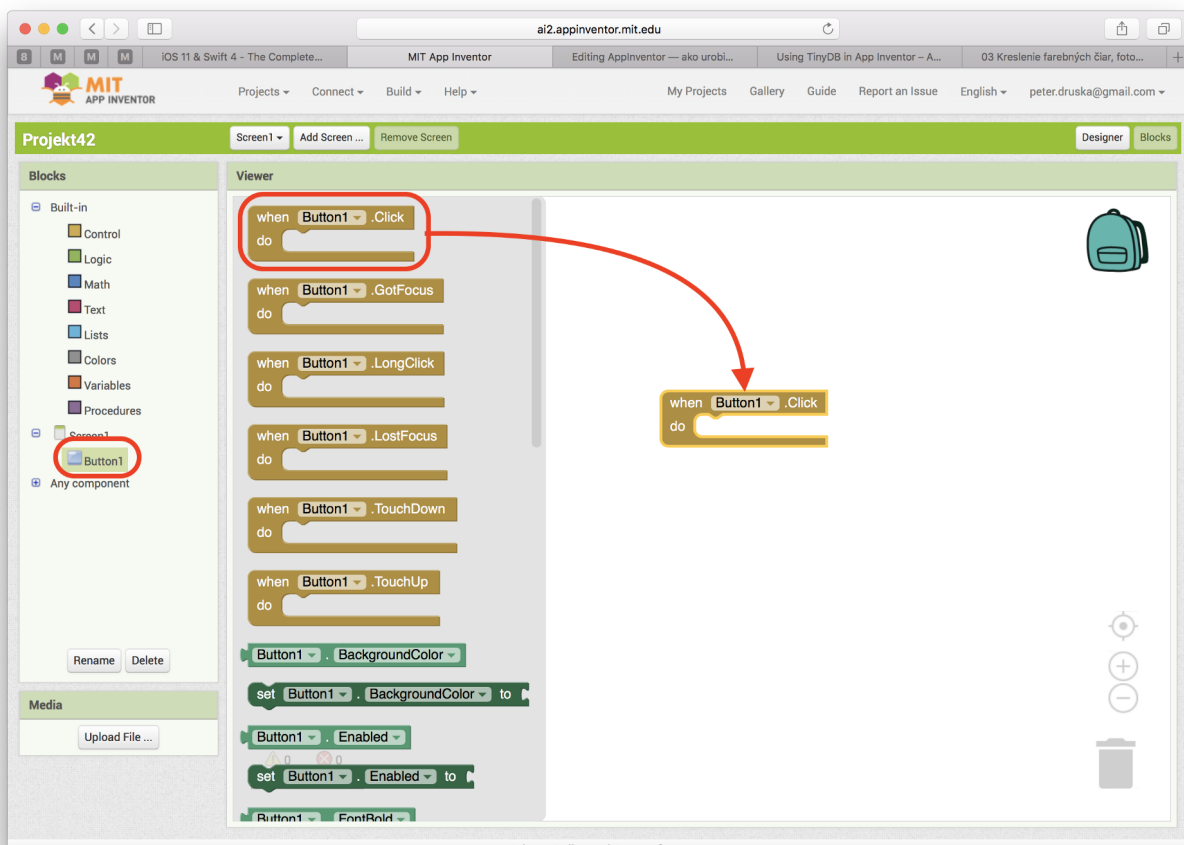


**Obrázok 9.** Blocks: časť určená na programovanie.

Samotné programovanie tlačidla funguje tak, že ho označíme, pričom sa automaticky zobrazia príkazy, ktoré zodpovedajú funkcii tlačidla. Napríklad:

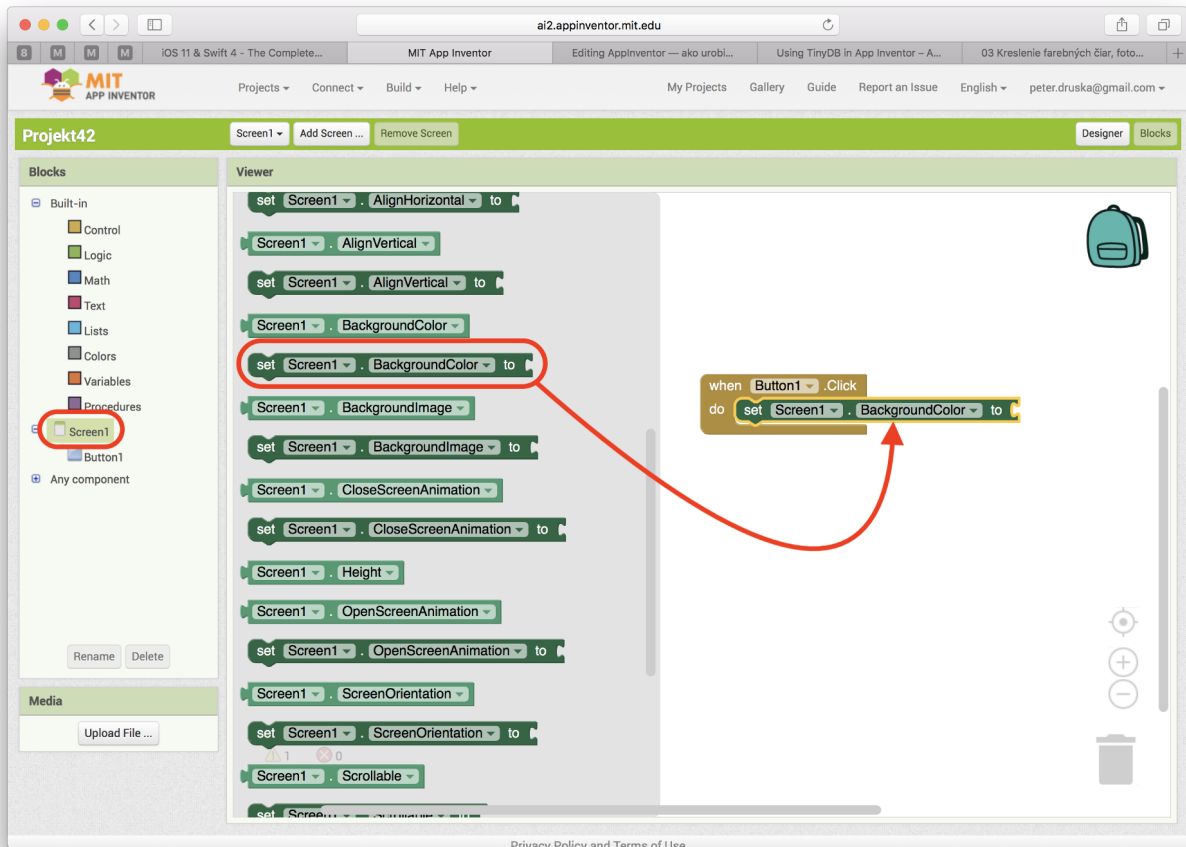
- Keď je tlačidlo kliknuté,
- Keď je tlačidlo označené,
- Keď sa tlačidla dotkne,
- Nastav tlačidlo text,
- ...

Je tu nutná istá znalosť základnej angličtiny, ale nič, čo by sa každý nevedel naučiť. Takže ak chceme, aby tlačidlo menilo farbu pozadia celej aplikácie, tak pridáme najskôr príkaz **When Button1.Click do**. Označíme tlačidlo **Button1** v časti Blocks a príkaz na kliknutie jednoducho potiahneme do hlavnej plochy.



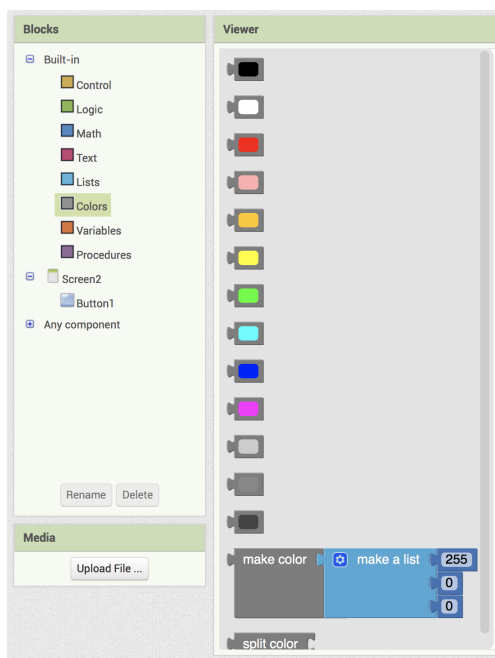
**Obrázok 10.** Príkaz čakajúci na kliknutie na tlačidlo.

Následne do príkazu na kliknutie pridáme zmenu farby pozadia tak, že v ľavej časti označíme **Screen** (obrazovku) a z ponuky príkazov pridáme príkaz **Set Screen.BackgroundColor** to.



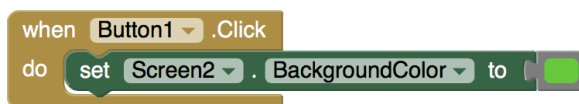
**Obrázok 11.** Príkaz, ktorý po stlačení tlačidla (hnedý) zafarbí obrazovku na (zelený) danú farbu.

Ešte ostáva vybrať farbu. V ľavej časti klikneme na položku **Colors**.



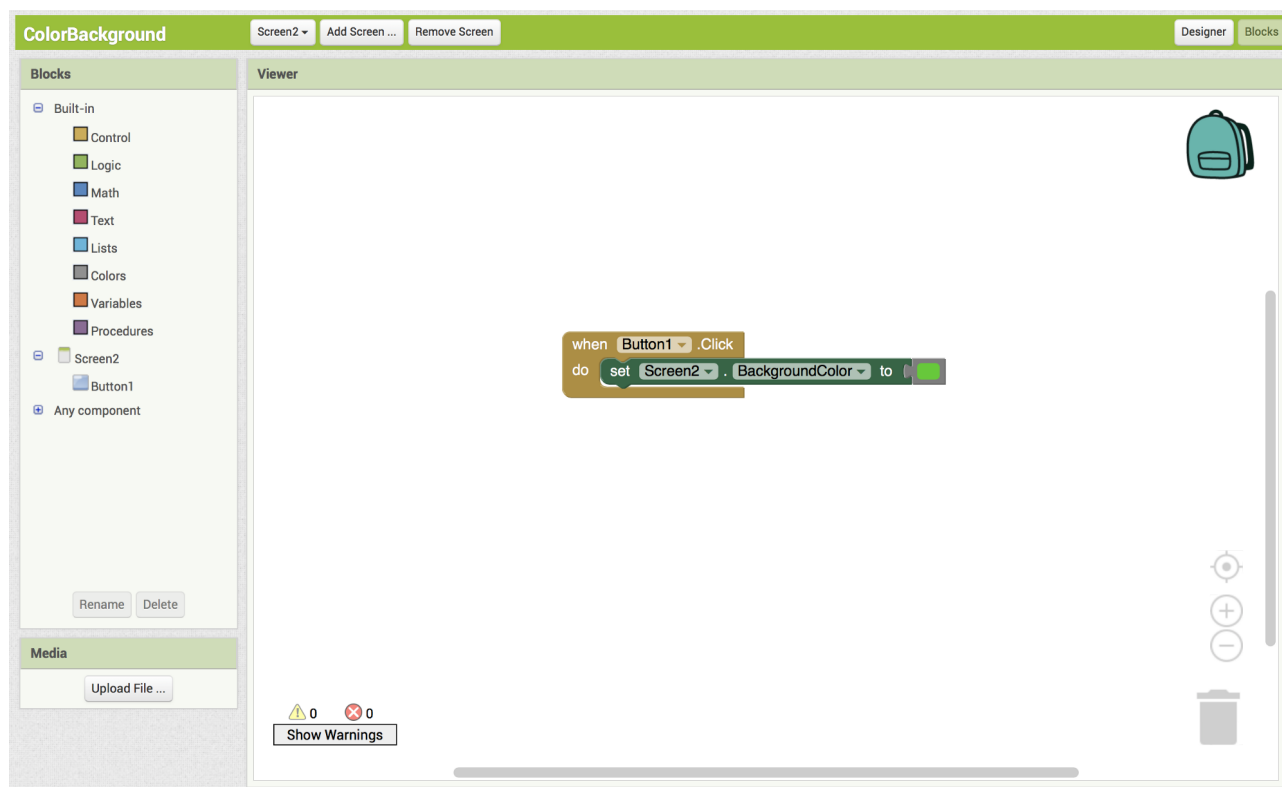
**Obrázok 12.** Výber farebných hodnôt do príkazu.

Následne jednu z farieb pridáme do nášho príkazu na stlačenie tlačidla.



**Obrázok 13.** Nastavenie zelenej farby celej obrazovky po stlačení tlačidla.

Celá obrazovka naprogramovaného tlačidla vyzerá takto.

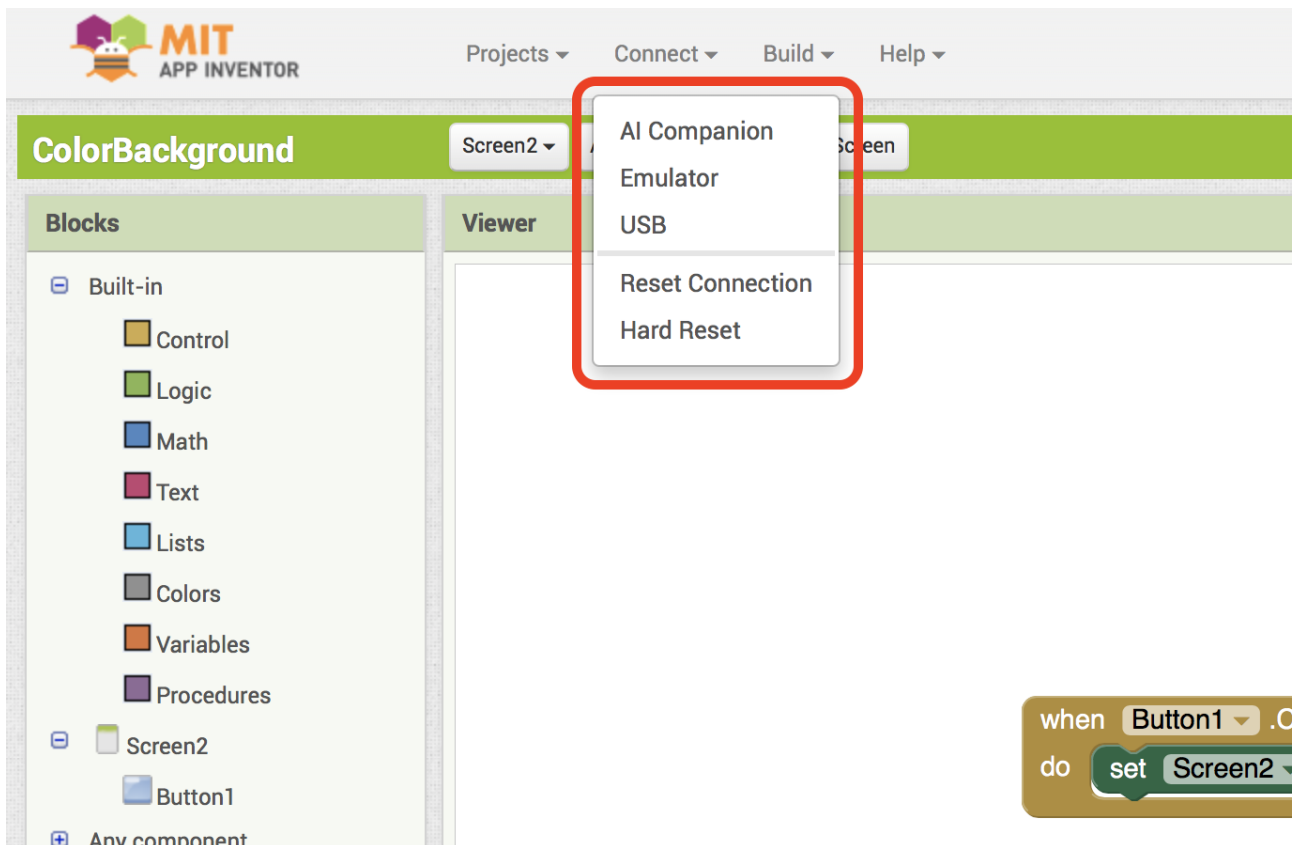


**Obrázok 14.** Naprogramované tlačidlo na zmenu farby.

## Krok #4 – Pripojenie zariadenia

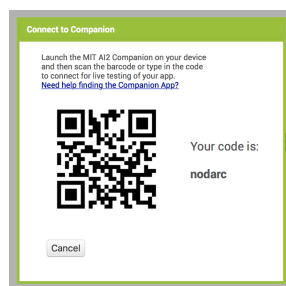
Verzia 1: Ak je počítač aj telefón na tej istej wifi sieti

V prvom rade je nutné do zariadenia nainštalovať appku MIT AI2 Companion, ktorú si zadarmo stiahneš z Google Play Store. Následne v rozhraní App Inventor otvoriť v menu časť **Connect** a vybrať ponuku **AI Companion**.



Obrázok 15. Výber typu pripojenia.

Následne sa otvorí okno s QR kódom. Ten je nutné cez stiahnutú appku zoskenovať. Ak nejde QR kód, tak treba manuálne prepísať textový kód.



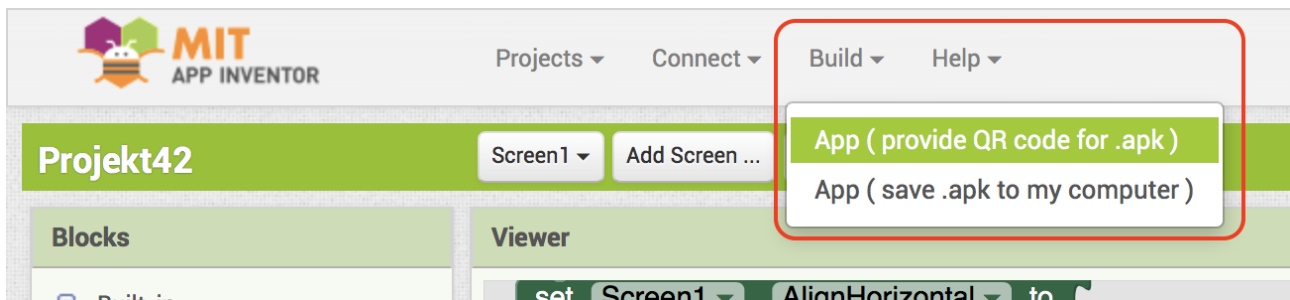
Obrázok 16. QR a textový kód.

Tento postup funguje len ak je počítač aj telefón na tej istej wifine.

## Verzia 2: Ak počítač a telefón nie sú na tej istej wifi sieti

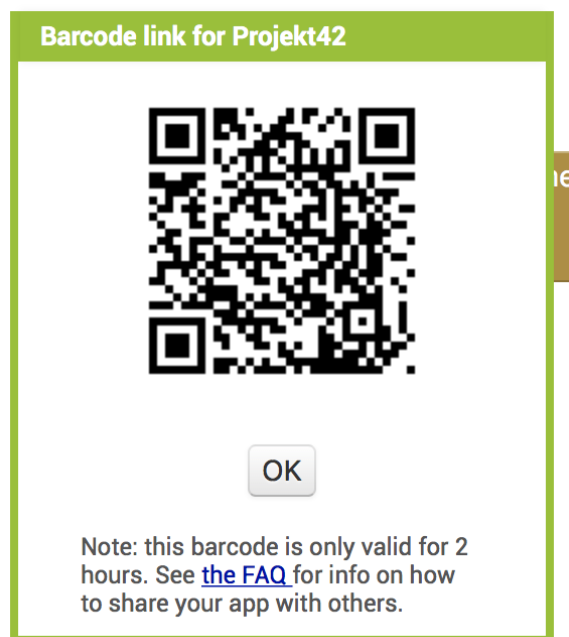
Tu je nutné, aby telefón mal prístup na internet, či už cez wifi sieť, alebo mobilné dáta.

Počítač môže byť na inej sieti alebo na kábli. V takom prípade v menu otvoríme časť **Build** a tam **App (provide QR code for .apk)**.



**Obrázok 17.** Vytvorenie APK súboru pre inštaláciu do telefónu.

Keď prebehne všetko, ako má, zobrazí sa QR kód.



**Obrázok 18.** QR kód na inštaláciu APK súboru aplikácie do zariadenia.

Po jeho zoskenovaní príslušnou appkou MIT AI2 Companion sa začne inštaláciu. Je možné, že telefón inštaláciu zablokuje, preto v nastaveniach Androidu je nutné povoliť

možnosť inštalácie z neznámych zdrojov (<https://techbrain.cz/2017/01/android-zaklady-telefonu-povolit-instalaci-neznamych-zdroju/>).

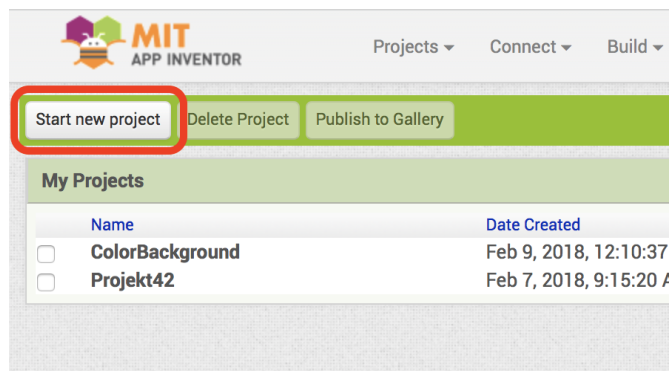
## **Krok #5 – Je na tebe**

Kam túto appku posunieš, je na tebe. Môžeš urobiť to, že tlačidlo raz pozadie zafarbí, inokedy nie (použitie premenných, ang. **variables**). Alebo bude prepínať náhodne (ang. **random**) farby. Alebo bude robiť niečo iné. Je to na fantázii.

Chod' do toho!

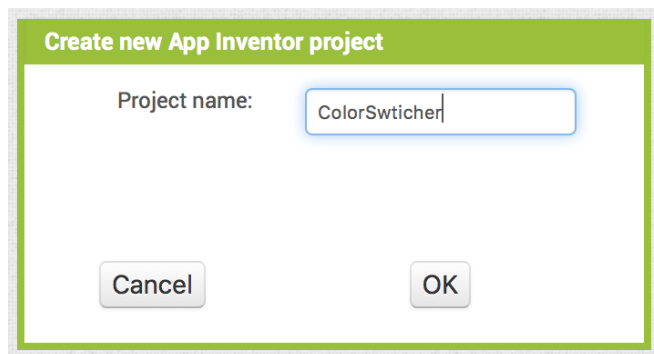
# Appka: prepínač farieb

Ako prvé treba vytvoriť nový projekt.



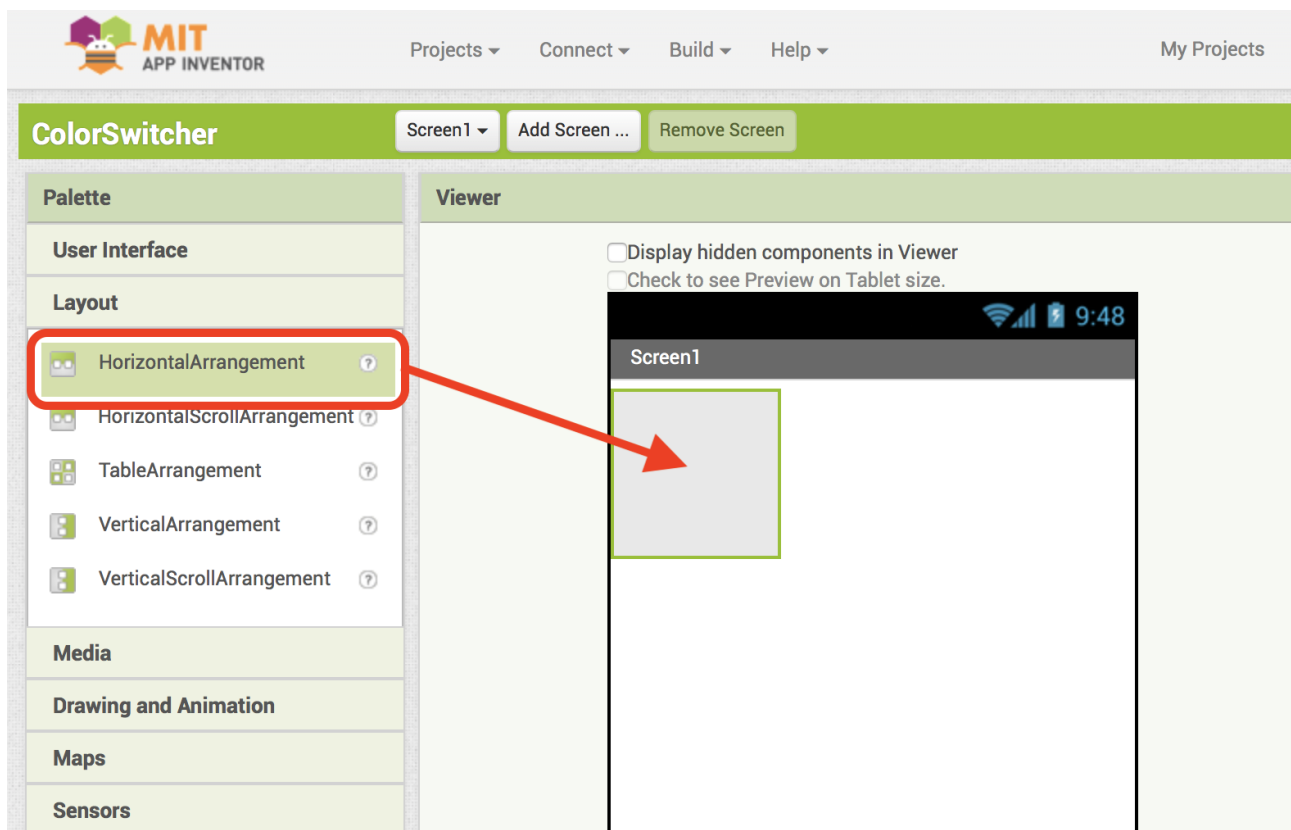
**Obrázok 1.** Start new project.

Potom je nutné zadať meno projektu. Bude to prepínač farieb, preto volím **ColorSwitcher**.



**Obrázok 2.** Meno projektu: ColorSwitcher.

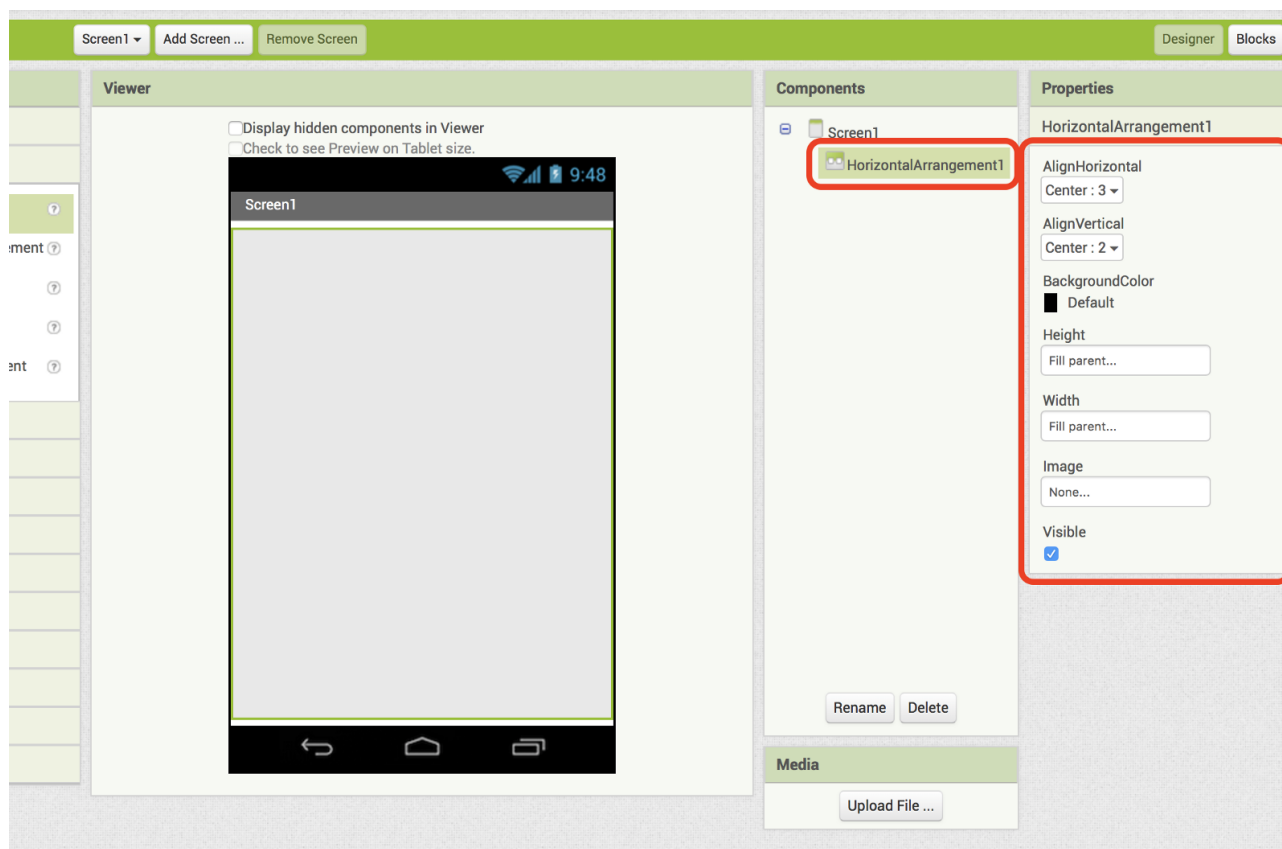
Potom daný projekt otvoríme a do obrazovky potiahneme **HorizontalArrangement**.



**Obrázok 3.** HorizontalArrangement.

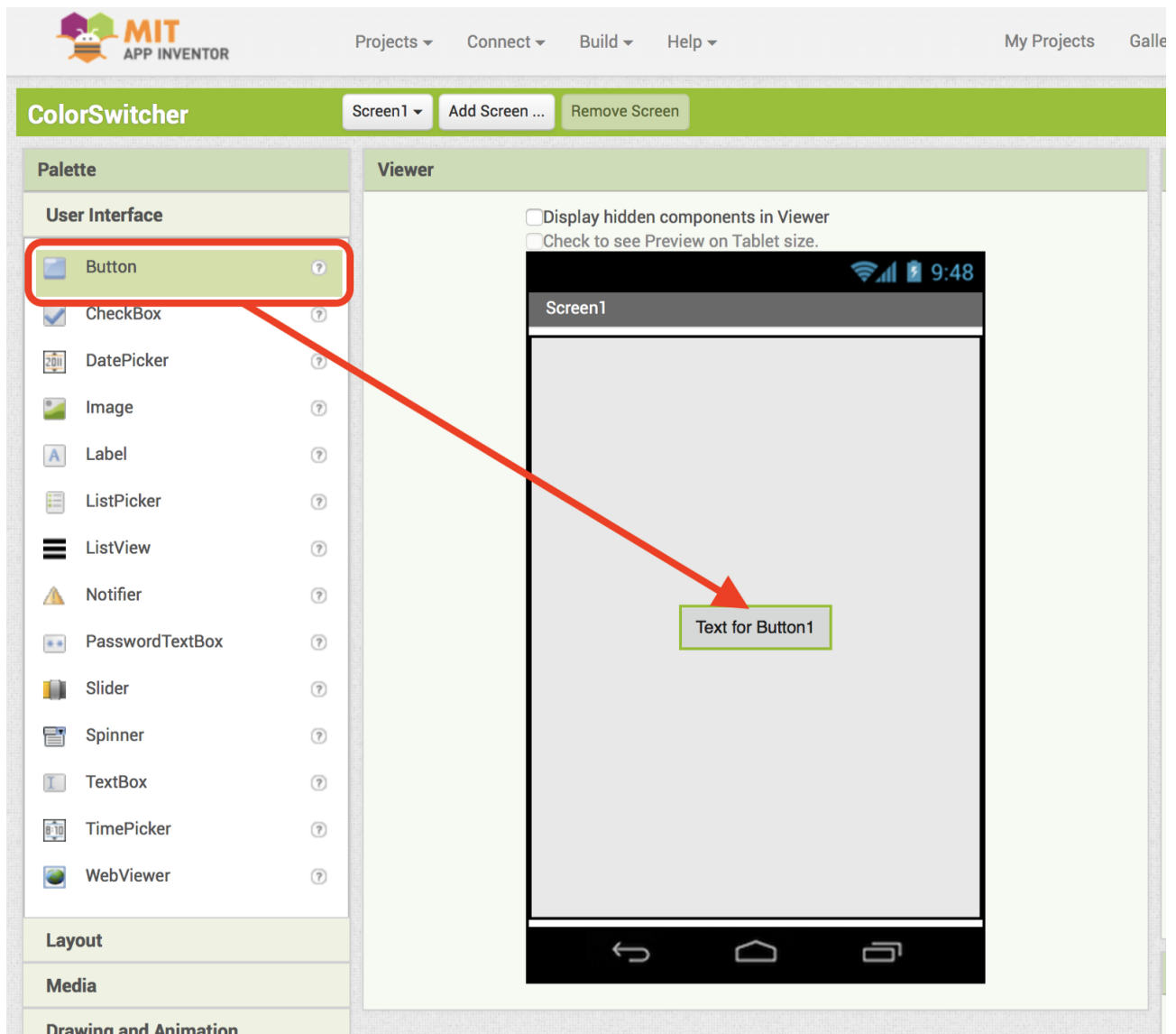
Je to objekt, ktorý všetky prvky v sebe vizuálne zarovná tak, aby dobre bolo. Na to mu musíme nastaviť isté vlastnosti v časti **Properties**.

Konkrétne horizontálne zarovnanie na stred (**AlignHorizontal: Center**), zarovnanie vertikálne na stred (**AlignVertical: Center**), výšku tak, aby vyplnila celé okno (**Height: Fill Parent**), šírku tak, aby vyplnila celé okno (**Width: Parent**).



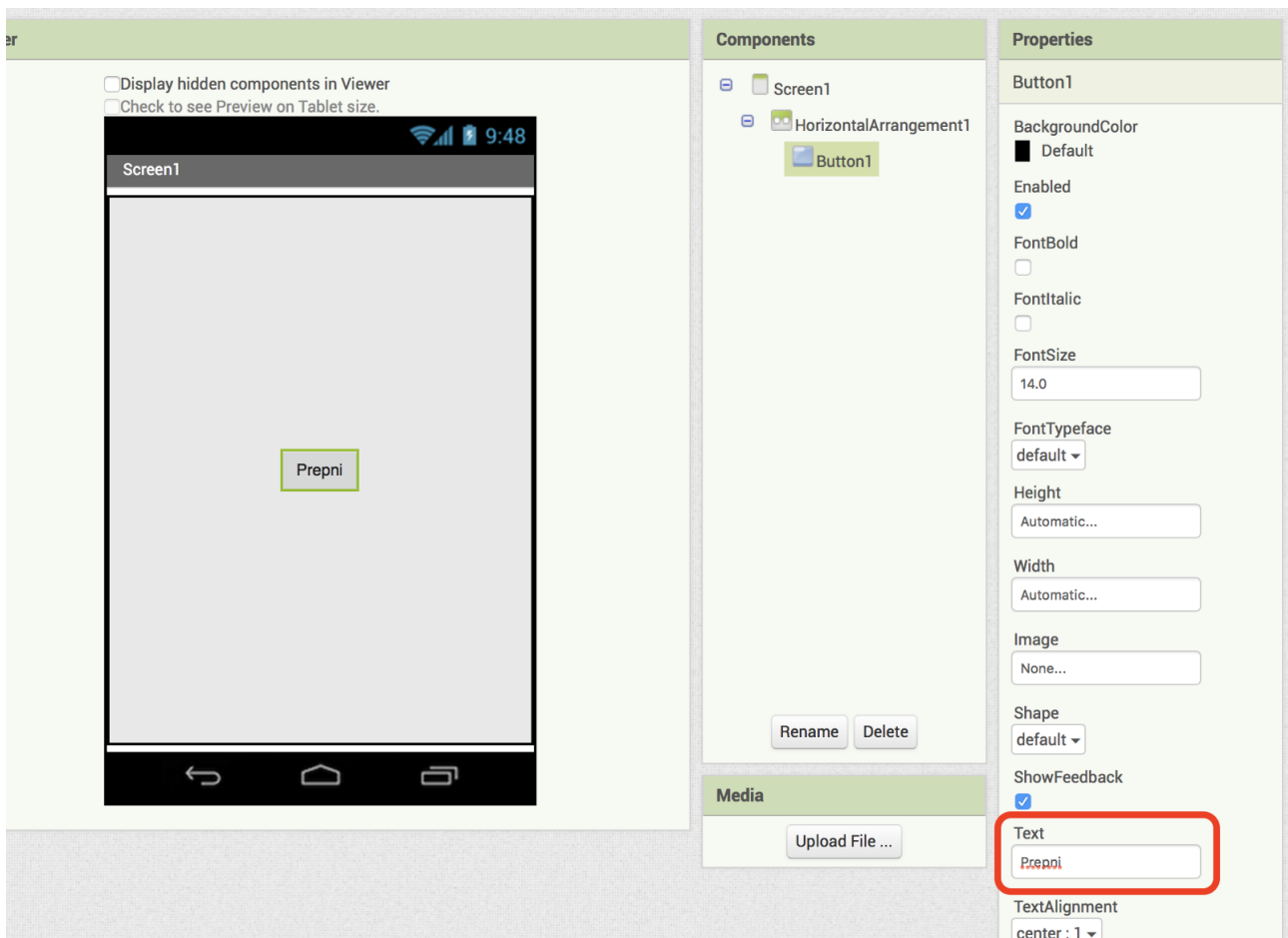
Obrázok 4. Zarovnávanie.

Keď teraz do tohto objektu potiahneme tlačidlo (**Button**), tak bude zarovnané horizontálne i vertikálne na stred. Inak povedané, bude v strede obrazovky.



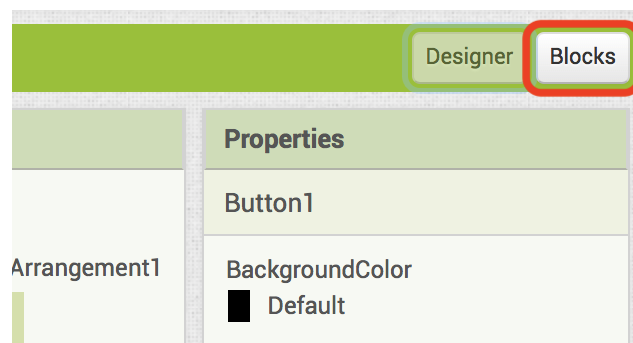
**Obrázok 5.** Pridanie tlačidla.

Keď tlačidlo označíme, tak v časti **Properties** zmeníme jeho vlastnosť **Text** napríklad na „Prepni“. Text tam môže byť ľubovoľný, no mal by dávať zmysel podľa toho, čo tlačidlo bude robiť.



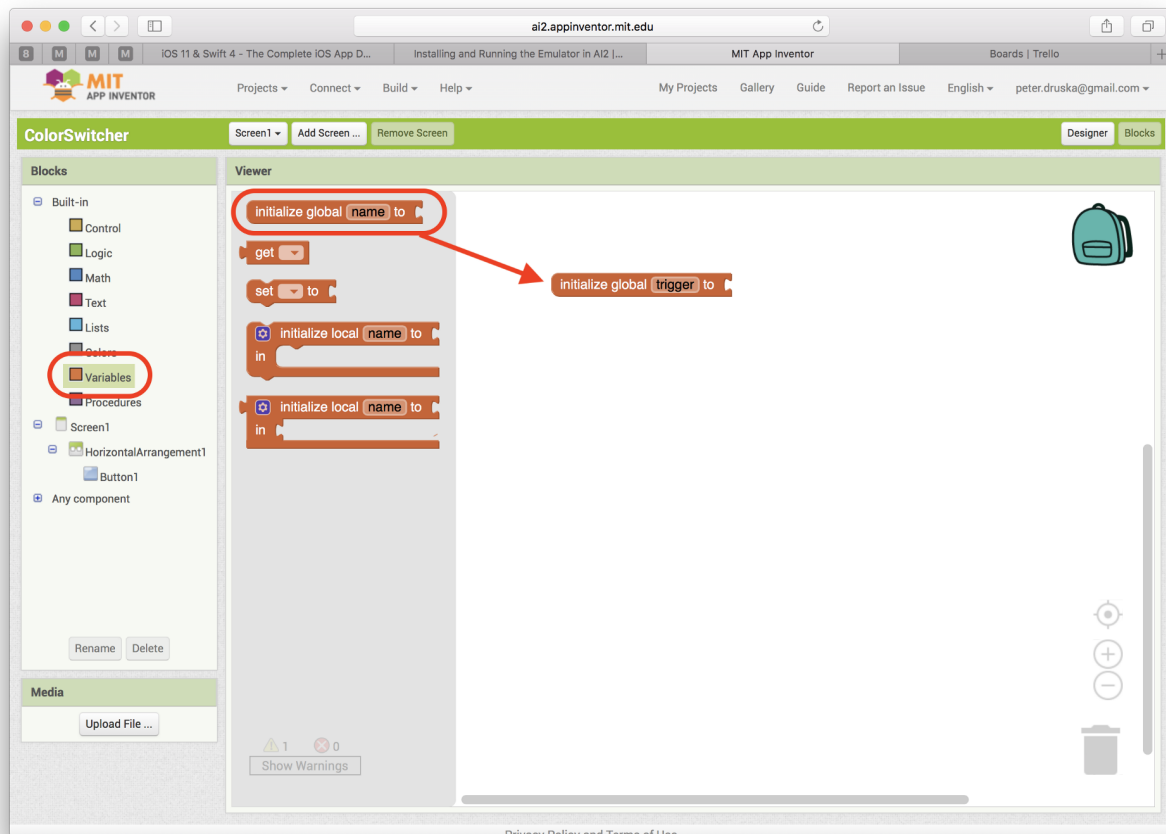
Obrázok 6. Zmena textu tlačidla.

Teraz sa prepneme do blokov (**Blocks**), ideme programovať pridané tlačidlo.



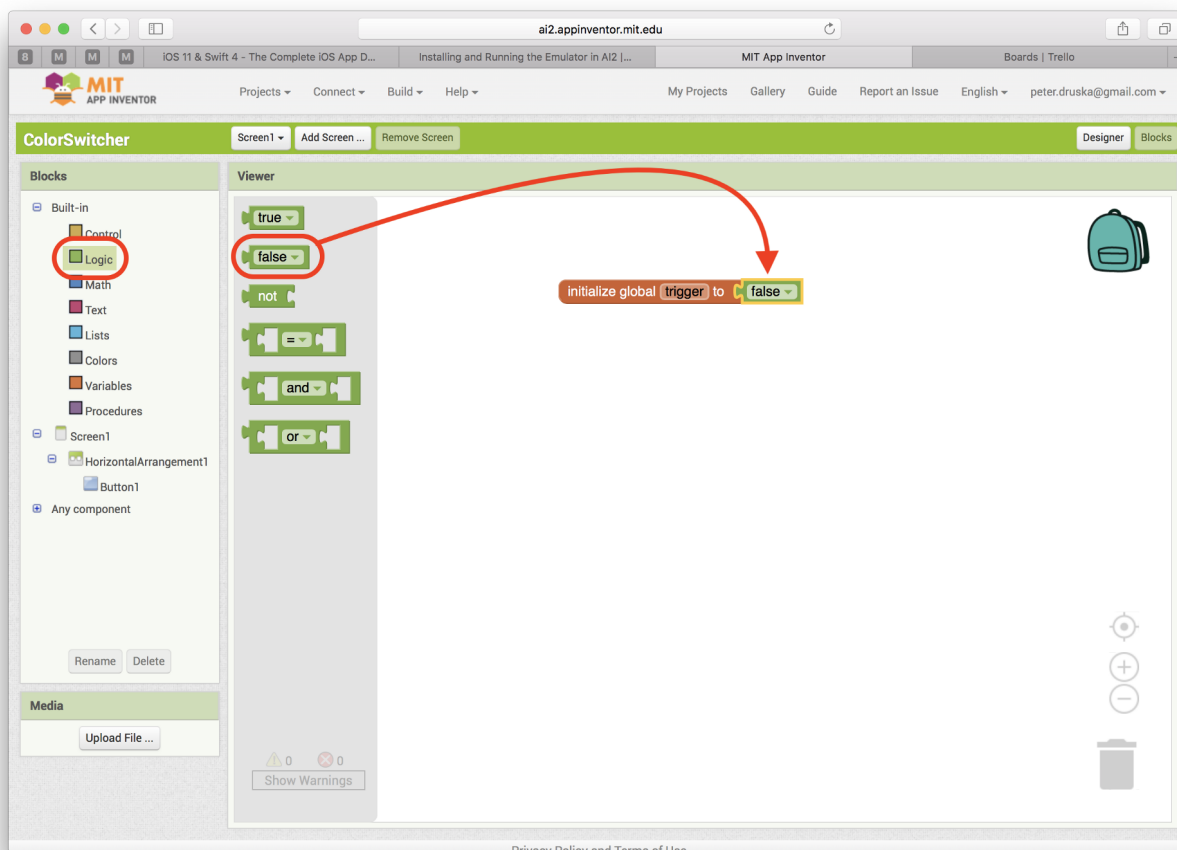
Obrázok 7. Prepnieme sa do blokov.

Ako prvú vytvoríme premennú, ktorá sa bude volať **trigger**. Bude to prepínač, ktorý bude zodpovedný za rozlíšenie toho, akú farbu máme zobrazíť.



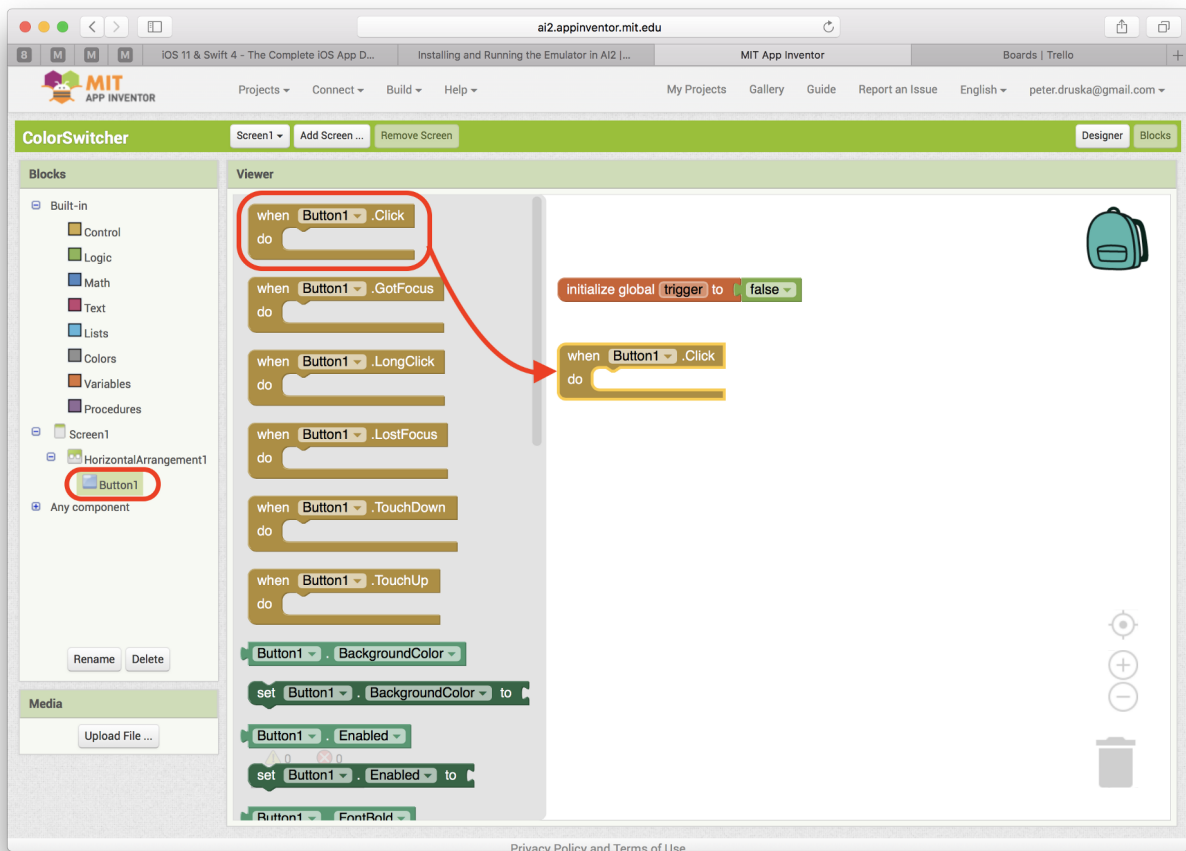
**Obrázok 8.** Definovanie premennej trigger.

Následnej premennej **trigger** musíme priradiť základnú hodnotu **false**, ktorú neskôr budeme meniť.



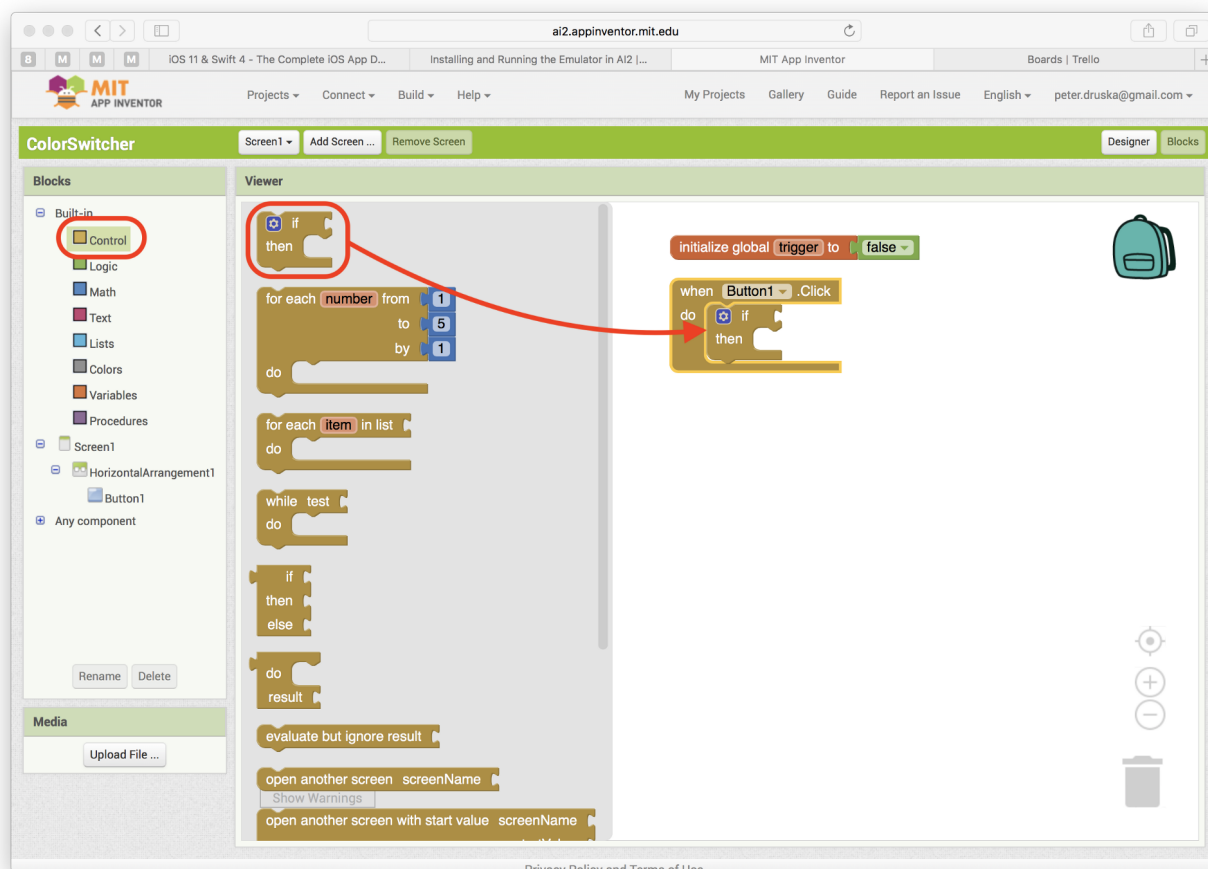
**Obrázok 9.** Priradenie hodnoty do premennej.

V ľavej časti označíme tlačidlo **Button1**, aby sme vyvolali zoznam udalostí, ktoré sa s tlačidlom viažu. Vyberieme **when Button1.Click do**. Inak povedané, keď sa na tlačidlo klikne/tapne, niečo sa vykoná.



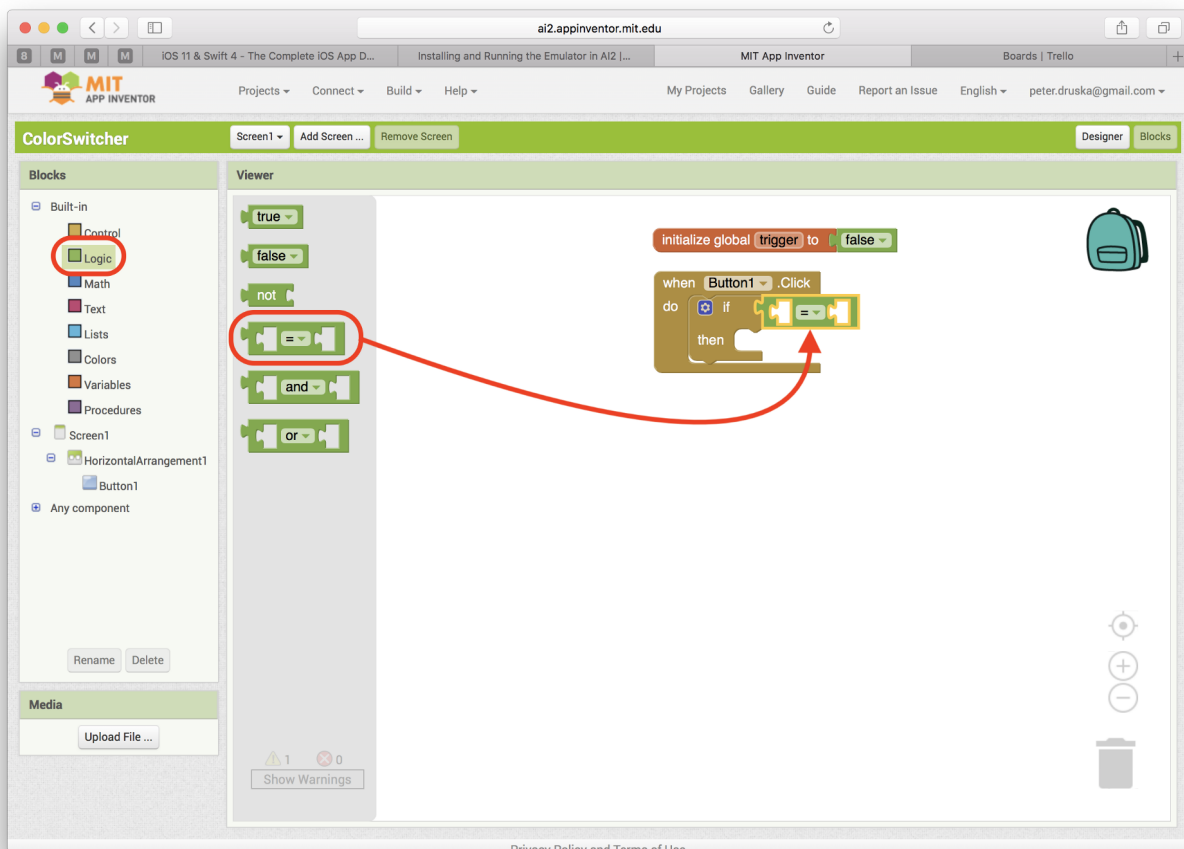
Obrázok 10. Pridanie udalosti, keď sa klikne na tlačidlo.

Nasleduje rozlíšenie stavu pomocou podmienky **if**. Toto rozlíšenie bude prebiehať v udalosti kliknutia na tlačidlo, preto **if** ťaháme do jej vnútra.



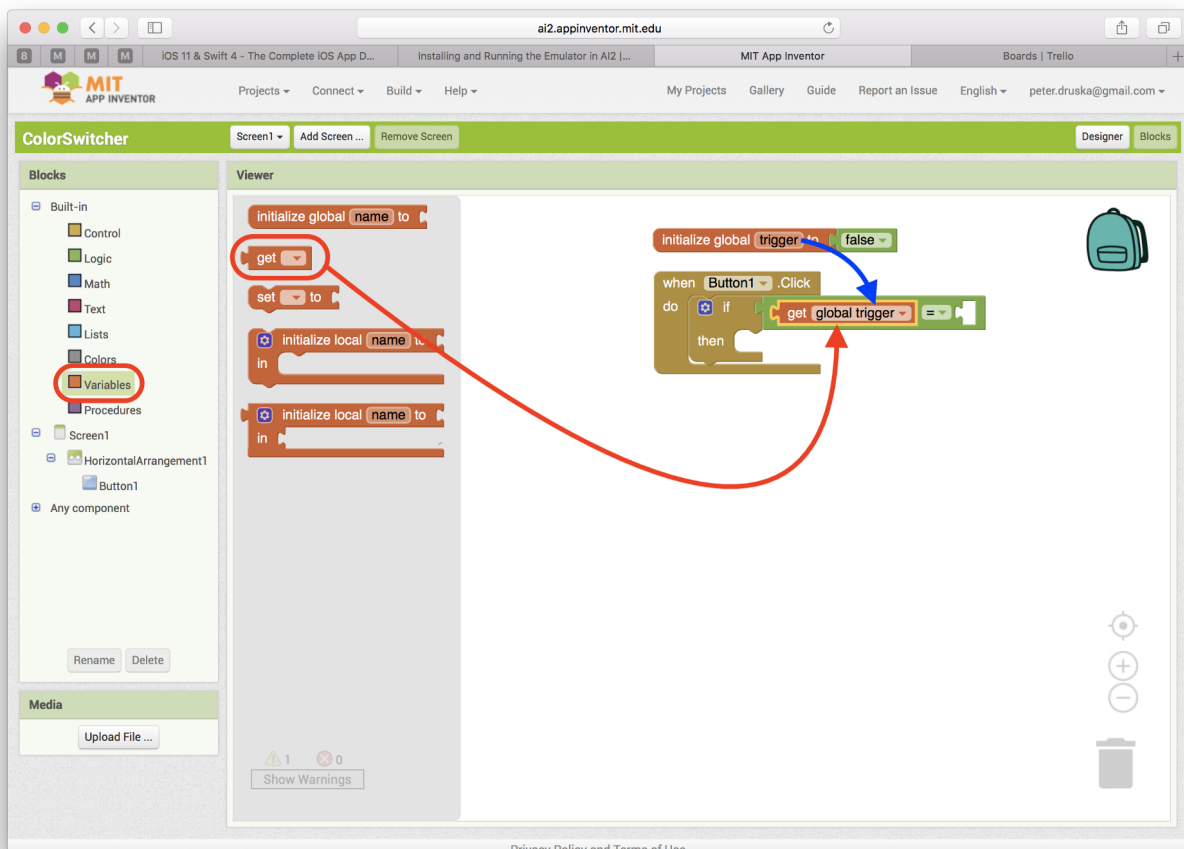
**Obrázok 11.** Pridanie rozhodovacieho bloku if.

Tento rozhodovací blok **if** musí o niečom rozhodnúť. My chceme porovnávanie. Preto logický blok porovnania potiahneme do podmienky **if**.



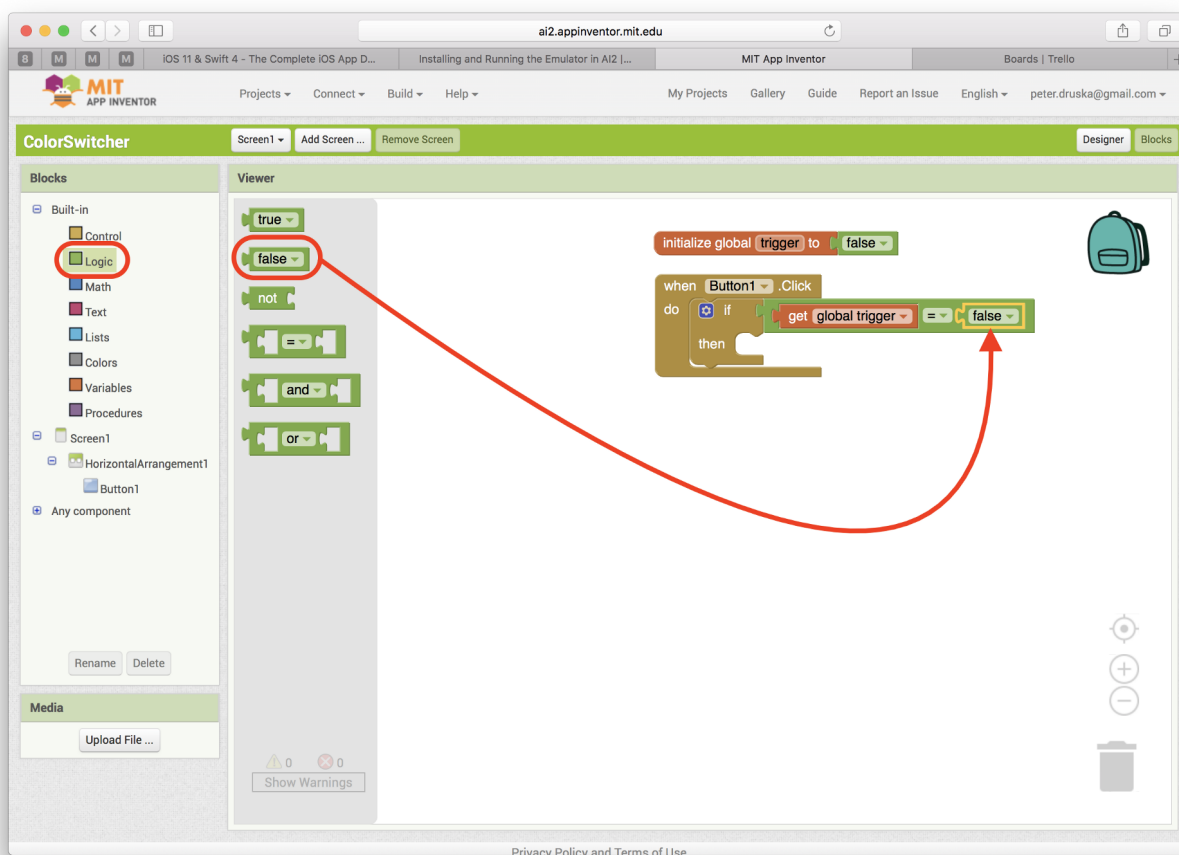
**Obrázok 12.** Pridanie porovnávacieho bloku do podmienky.

Porovnať chceme premennú **trigger**, preto si jej hodnotu vypýtame blokom **get global trigger** do ľavej časti porovnávania.



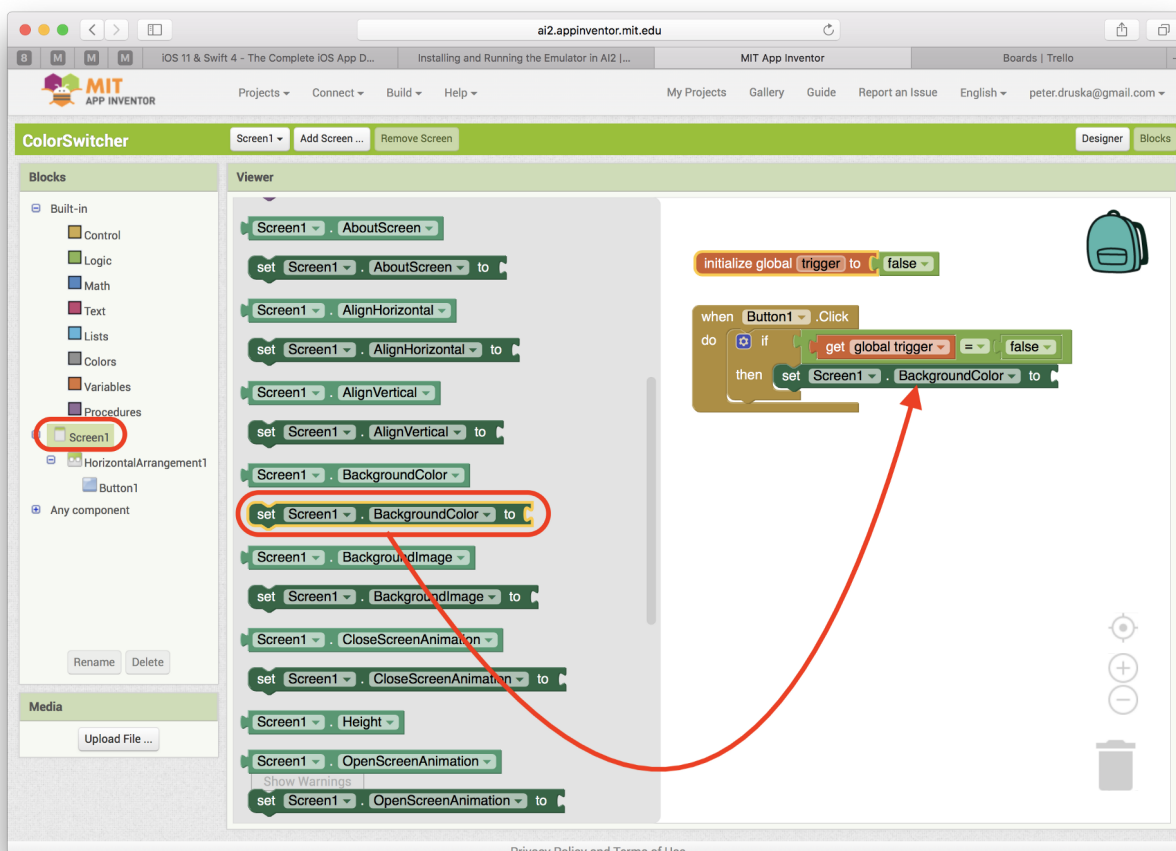
**Obrázok 13.** Pridanie hodnoty premennej trigger do porovnania.

Hodnotu premennej **trigger** porovnáme s logickou hodnotou **false**.



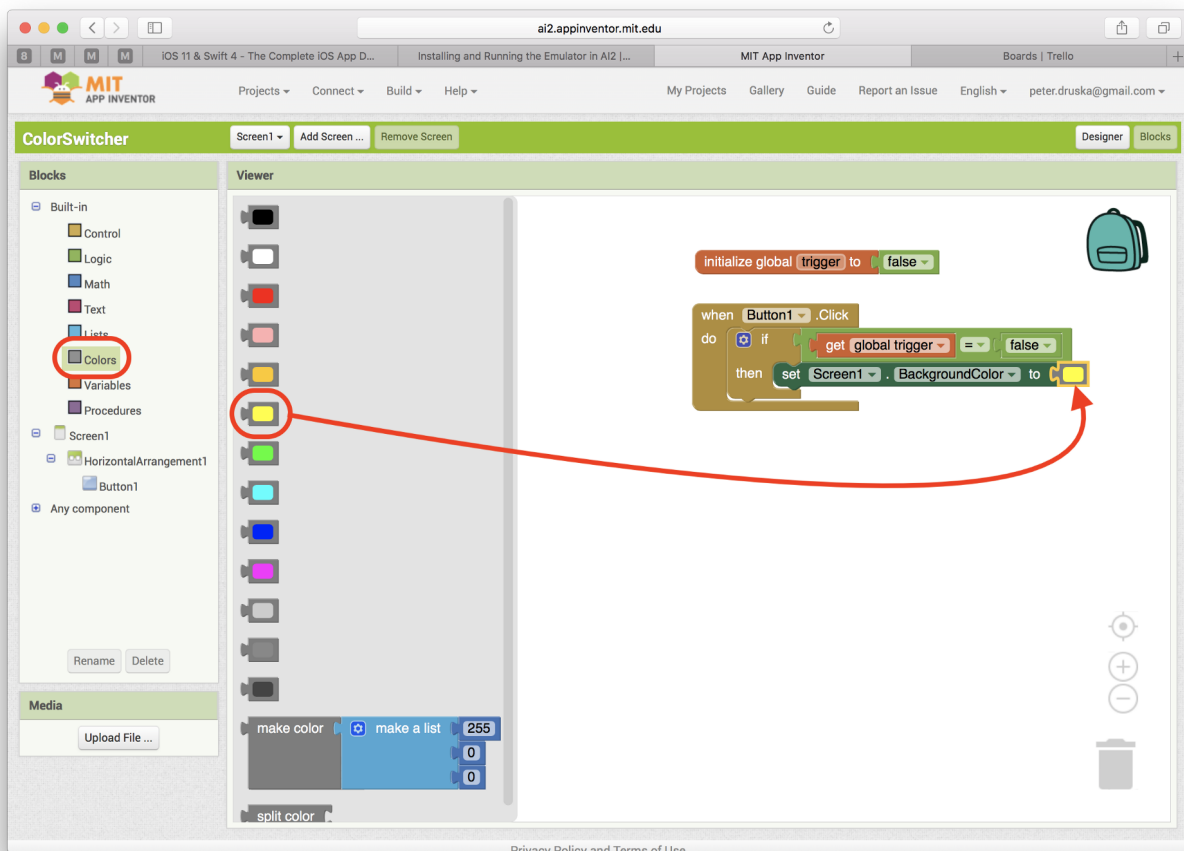
**Obrázok 14.** Porovnanie hodnoty premennej trigger s hodnotou false.

Táto podmienka sa bude pýtať niečo takéto: „*ak je hodnota premennej trigger false, tak (then) vykonaj NIEČO*“. To niečo bude zafarbenie pozadia celej obrazovky. Vyberieme blok **Screen1** a blok **set Screen1.BackgroundColor** to potiahneme do podmienky **if**.



**Obrázok 15.** Pridanie bloku na nastavenie farby pozadia celej obrazovky.

Ďalej vyberieme farbu, ktorá sa má zapnúť, keď sa tento blok príkazov vykoná. V príklade som použil žltú, neváhaj, a skús si ju zmeniť na inú.

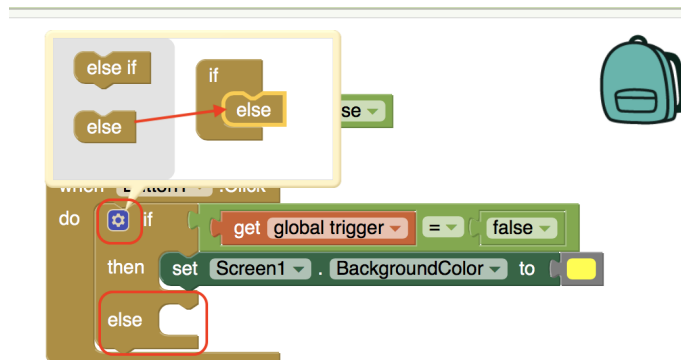


**Obrázok 16.** Pridanie farby pozadia.

Skús toto vo svojom zariadení. Jediné, čo to bude robiť, že ofarbí pozadie obrazovky na žltú farbu a viac nič. Žltá farba tam ostane neustále, nech urobíš čokoľvek ďalšie v appke. Teraz si ukážeme, ako pozadie meniť na inú farbu podľa stlačenia tlačidla.

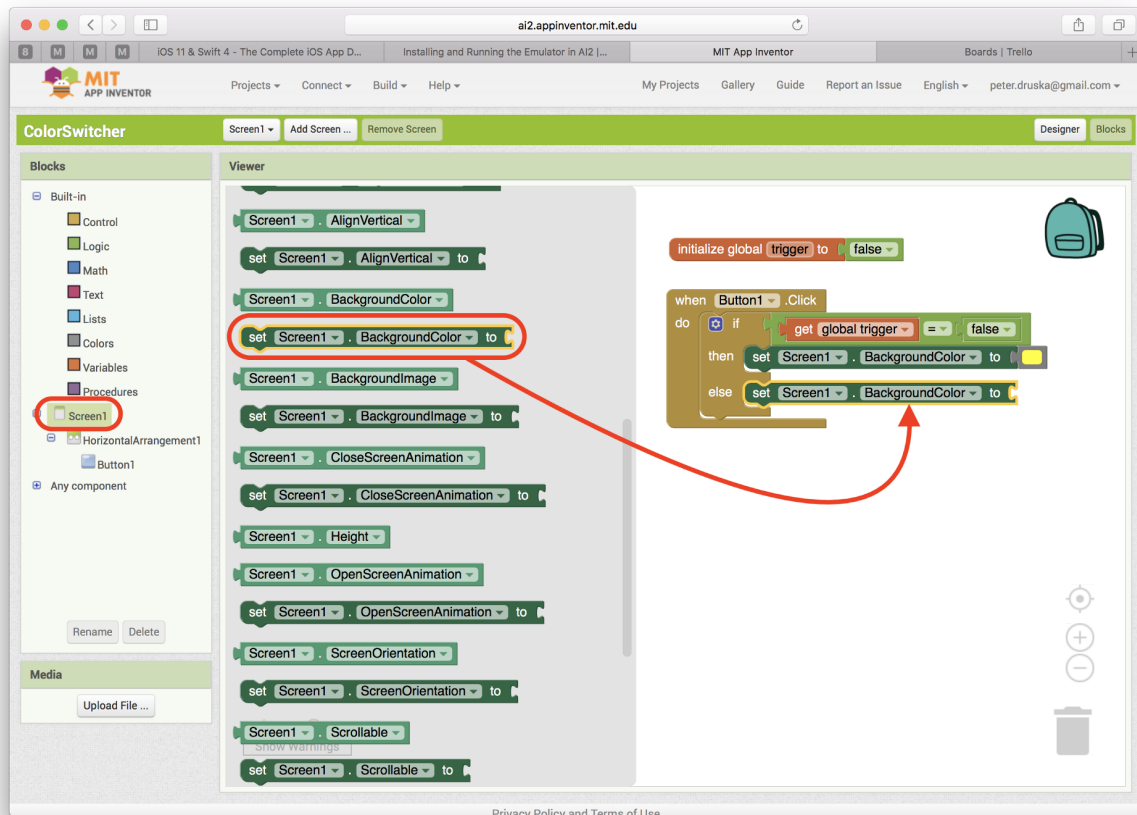
Pri podmienke **if** je modré ozubené koliesko. Keď ho stlačíme, tak sa objaví možnosť pridať z ľavej časti vetvu **else** potiahnutím do podmienky **if** vpravo.

Týmto ťahom sa podmienka v hlavnom programe rozšíri na štruktúru **if then else**.



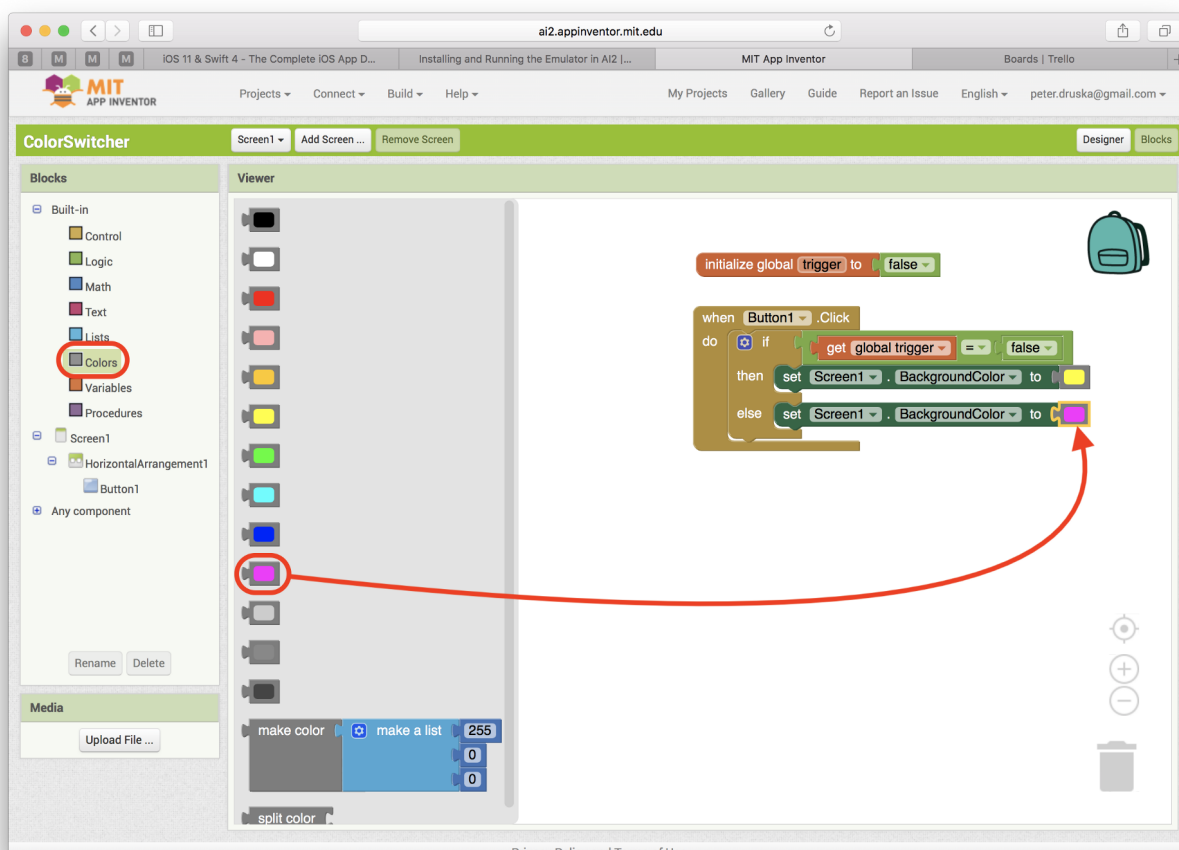
**Obrázok 17.** Rozšírenie podmienky o else vetvu.

Do else vetvy pridáme rovnaký blok na zmenu farby pozadia `set Screen1.BackgroundColor` to, ako predtým.



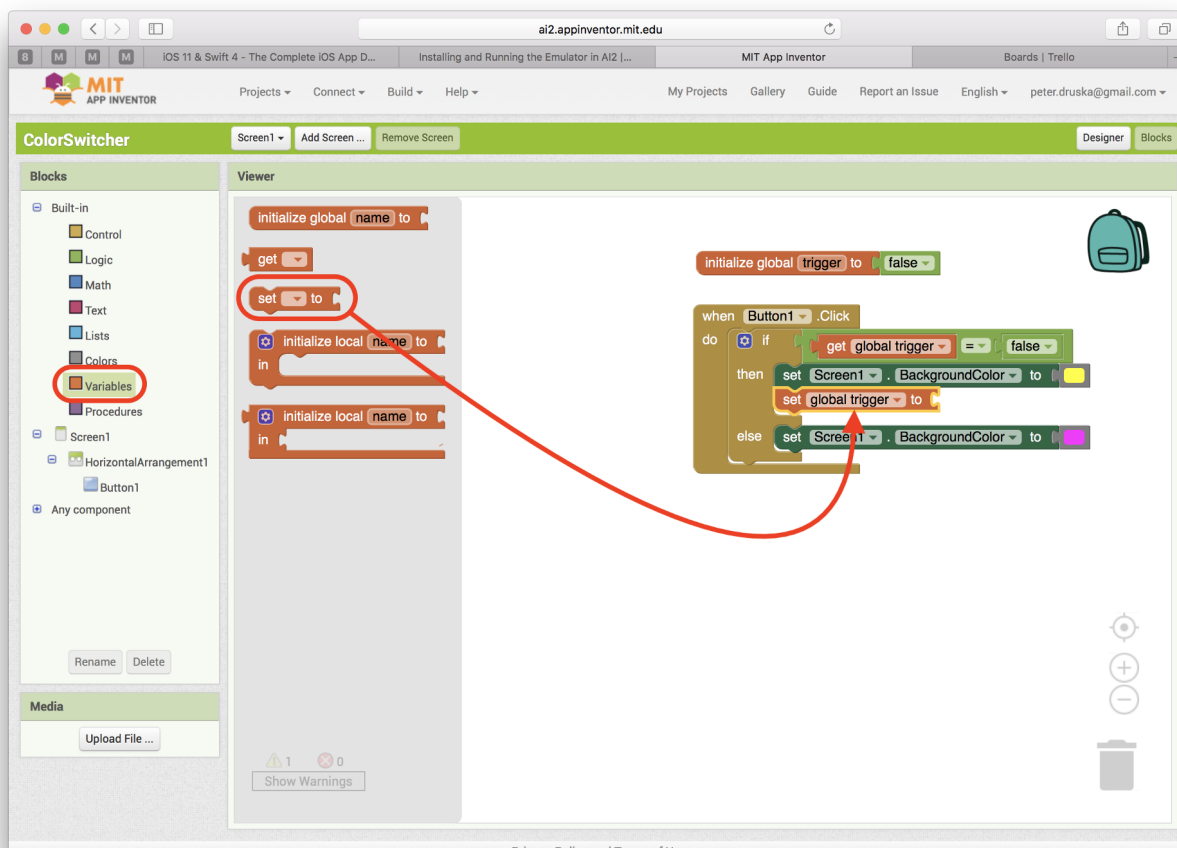
**Obrázok 18.** Zmena farby pozadia obrazovky v else vetve.

Ešte je nutné definovať farbu.



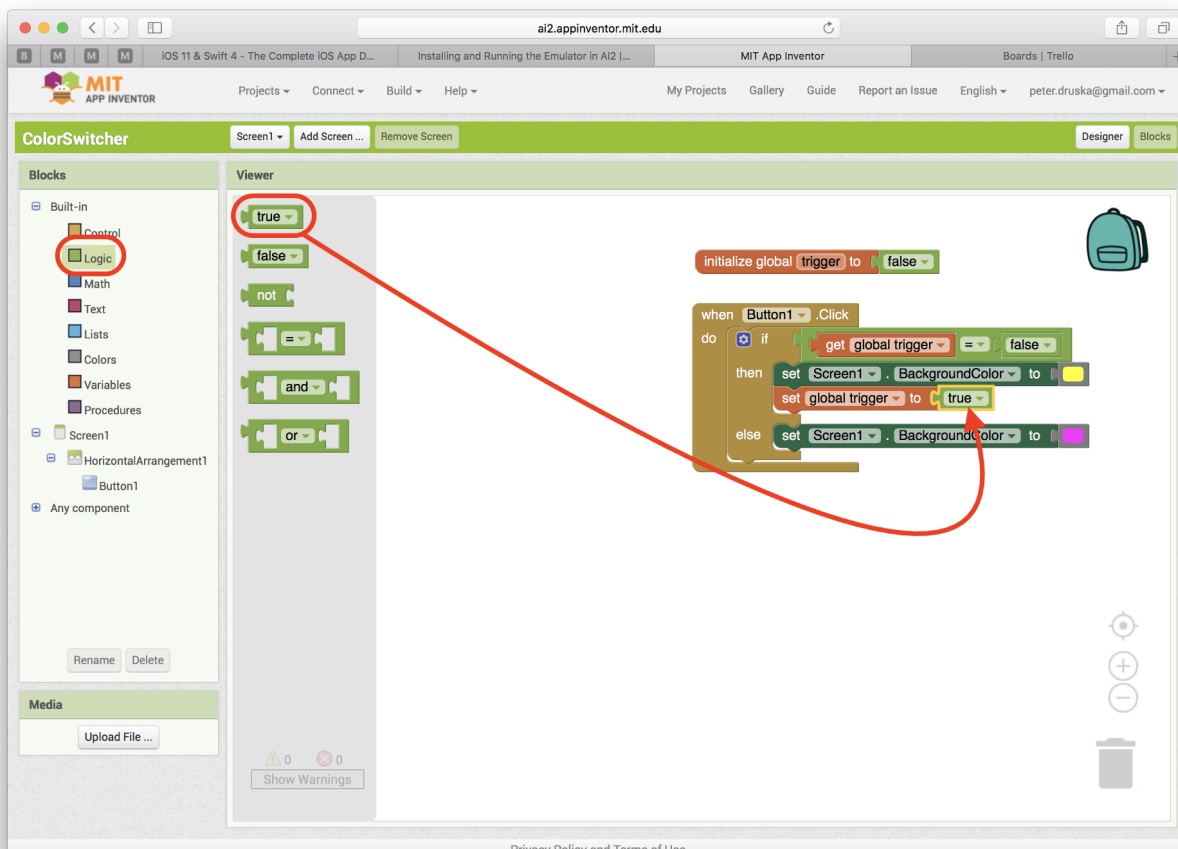
Obrázok 19. Pridanie konkrétnej farby.

Keď appku teraz spustíš na zariadení, nič sa nestane. Je to preto, že premenná **trigger** je stále nastavená na hodnotu **false** a tým pádom podmienka **if** nevbekne do vetvy **esle**. To sa stane vtedy, ak vnútri jednotlivých vetiev **if then else** pridáme zmenu hodnoty premennej **trigger** blokom **set global trigger to**.



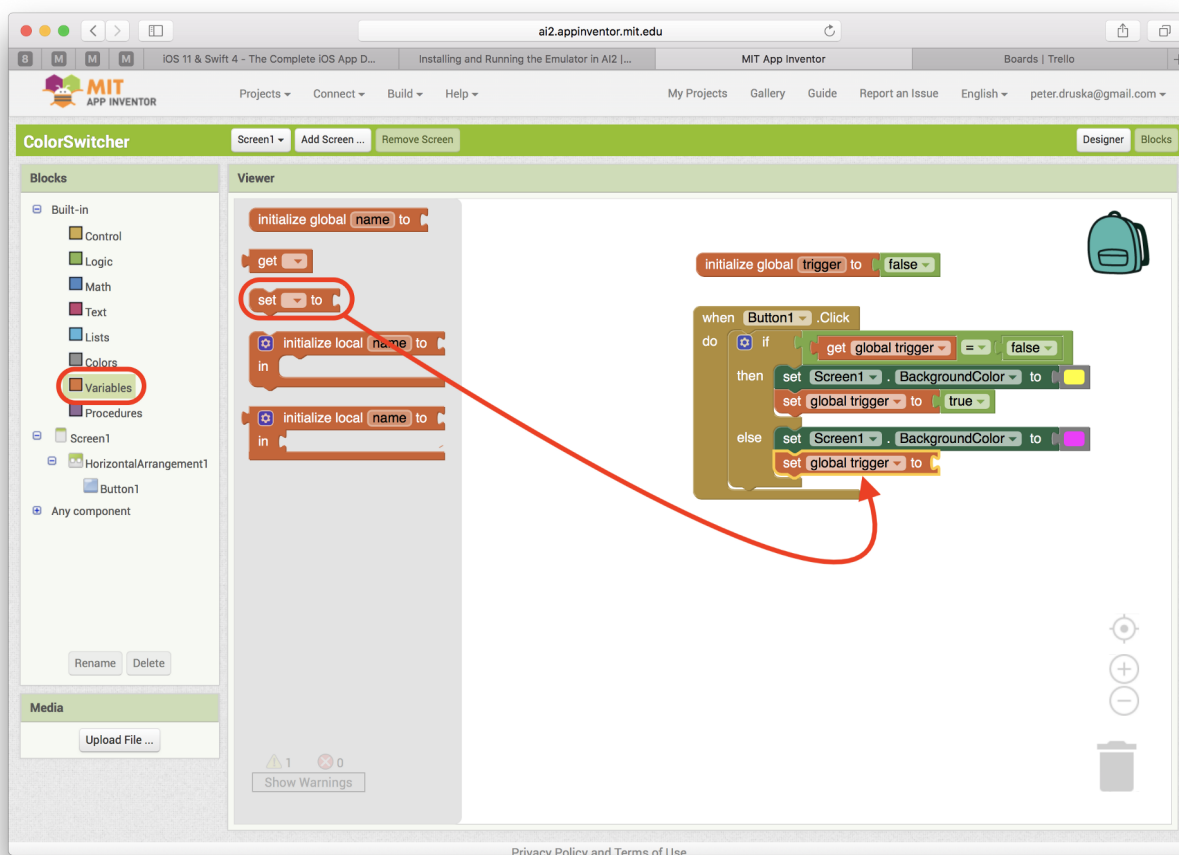
Obrázok 20. Nastavenie hodnoty premennej trigger.

Hodnotu premennej trigger nastavíme na **true**, keďže pri vstupe do tejto vetvy je **false**.  
Ako keby sme prepli svetlo: *ak nesvieti (false), tak zasviet (true)*.



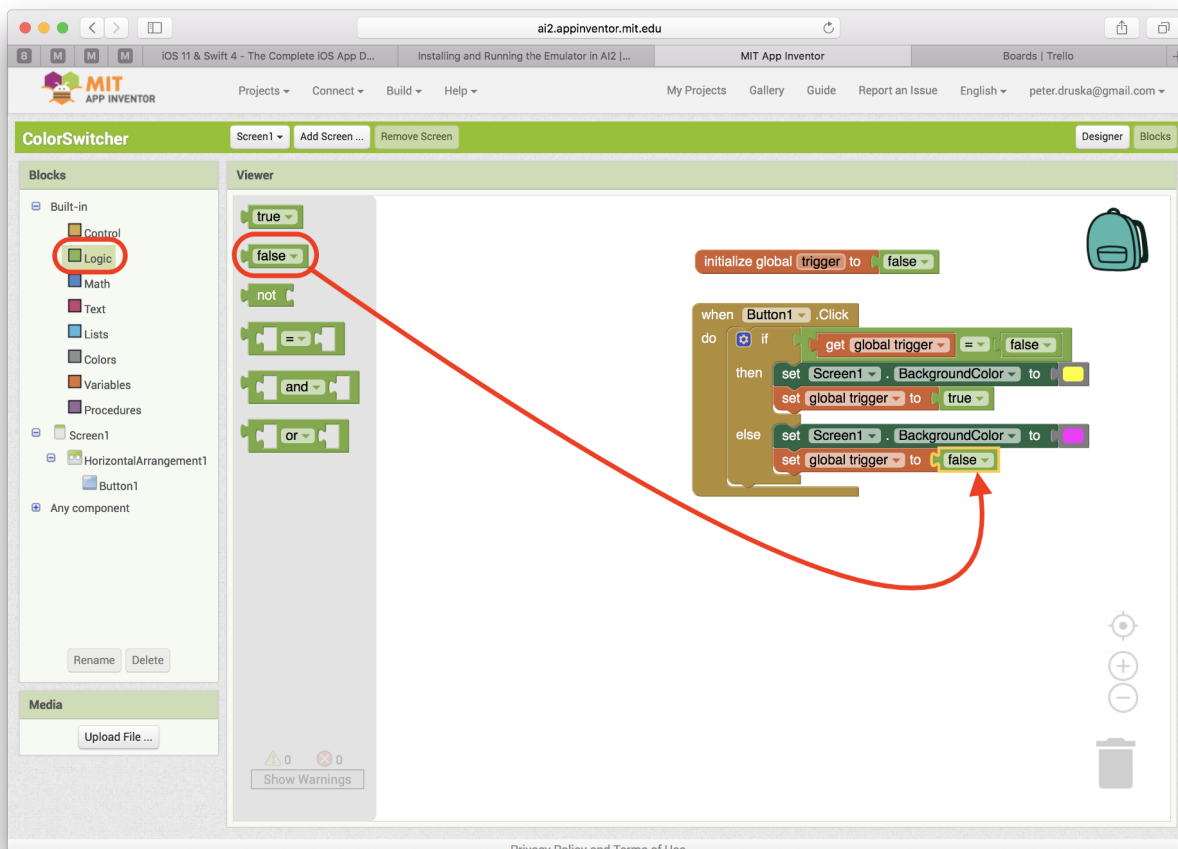
**Obrázok 21.** Prepnutie hodnoty premennej trigger na true.

To isté musíme urobiť vo vetve **else**.



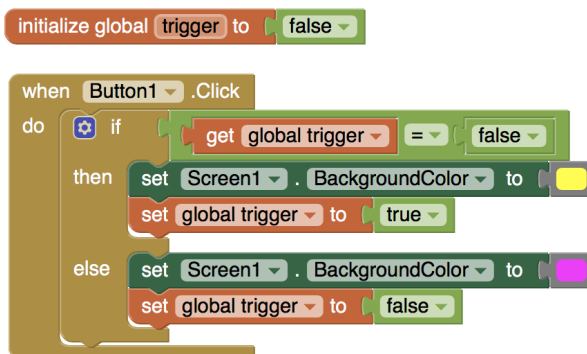
Obrázok 22. Nastavenie hodnoty premennej trigger vo vetve else.

Do vetvy **else** podmienky **if** vbehneme vtedy, ak hodnota premennej **trigger** je **true**. Preto ju tu nastavujeme naspäť na **false**.



**Obrázok 23.** Nastavenie hodnoty premennej trigger na false.

Celý program vyzerá takto.



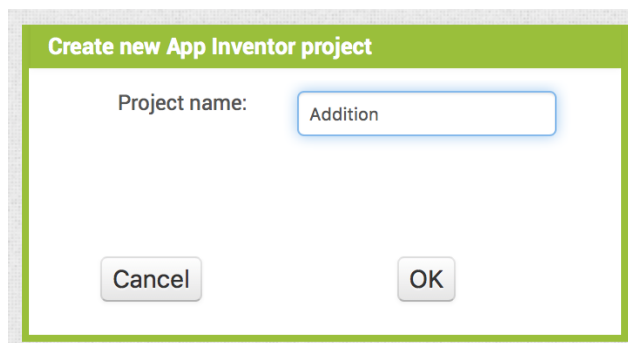
**Obrázok 24.** Celý program prepínača farby pozadia.

Keď teraz appku pustíš, opakovaním stláčaním tlačidla sa farba pozadia prepína zo žltej na ružovú a naspäť, a tak dookola.

Príjemné programovanie!

# Appka: sčítanie dvoch čísel

Vytvoríme nový projekt s názvom **Addition** (ang. súčet).

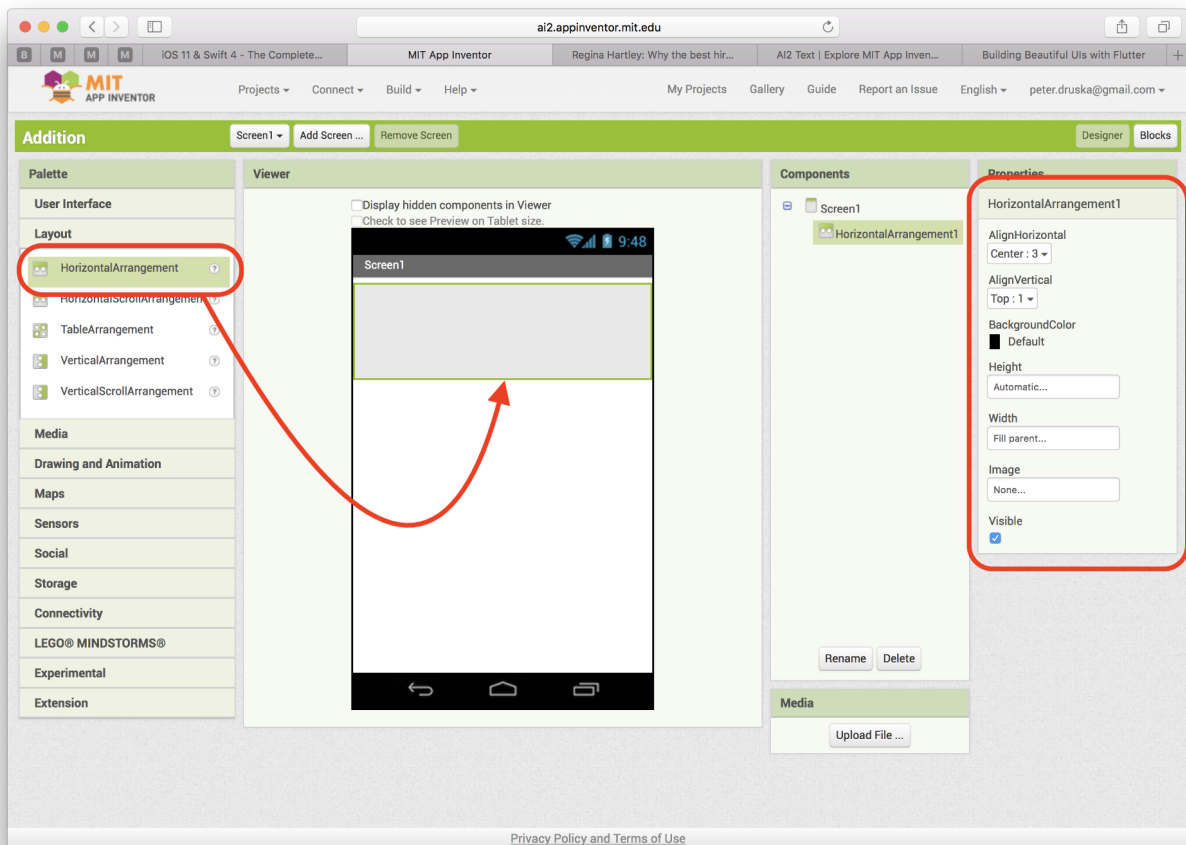


**Obrázok 1.** Názov projektu.

Do obrazovky **Screen1** potiahneme element **HorizontalArrangement**. Jemu nastavíme isté vlastnosti, ktoré vidíme na obrázku 2 v časti **Properties**:

- **AlignHorizontal: center**—horizontálne zarovnanie všetkého vo vnútri bude na stred.
- **AlignVertical: top**—vertikálne zarovnanie všetkého vo vnútri bude na horný okraj.
- **Height: Automatic**—výška sa bude prispôbovať obsahu.
- **Width: Fill Parent**—šírka sa prispôbí displeju.

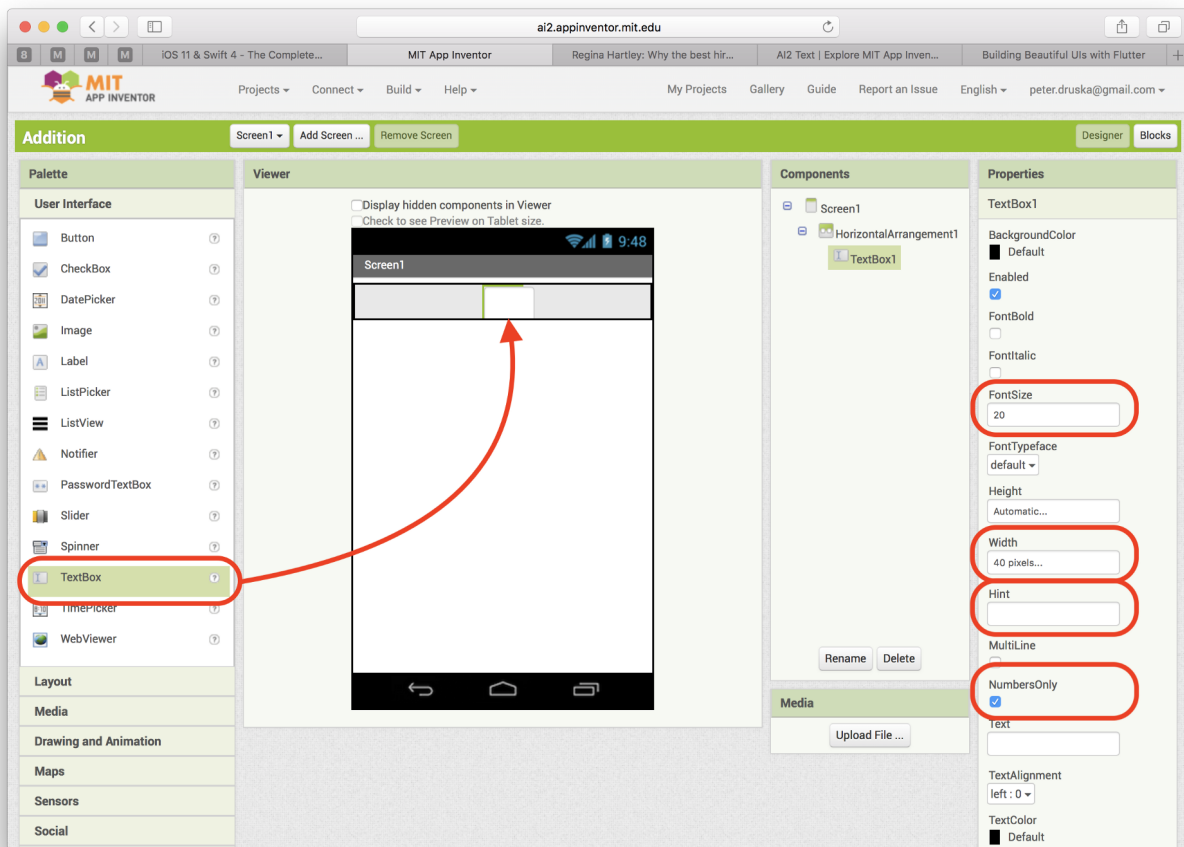
Zvyšok ponecháme tak, ako je.



**Obrázok 2.** Element horizontálneho usporiadania.

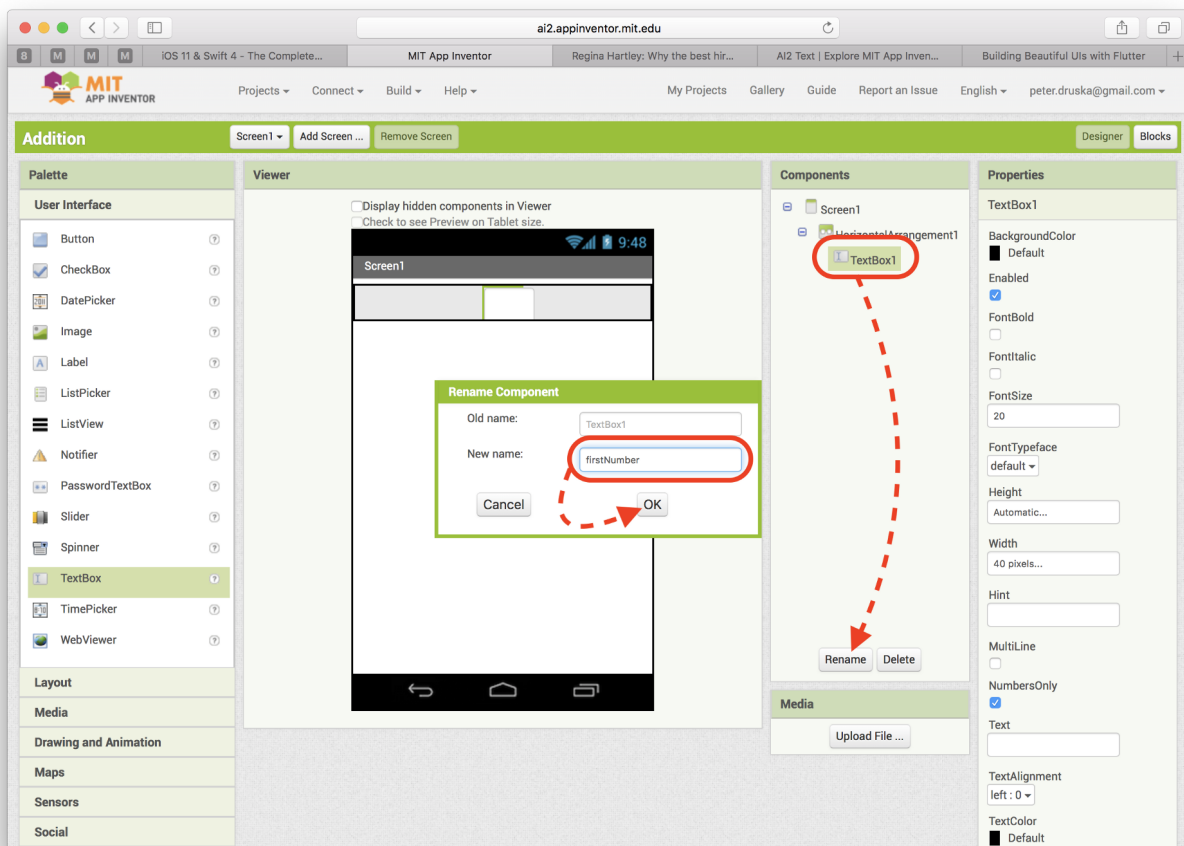
Do elementu na horizontálne usporiadanie pridáme **TextBox**, ktorému nastavíme tieto vlastnosti:

- **FontSize: 20**— veľkosť písma.
- **Width: 40 pixels**— šírka.
- **Hint: prázdne**— text, ktorý sa v textboxe bude zobrazovať ako nápoveda bude prázdny.
- **NumbersOnly: true/checked**— do textboxu sa bude dať zadať len číslo.



Obrázok 3. Pridanie textového políčka na zadanie textu.

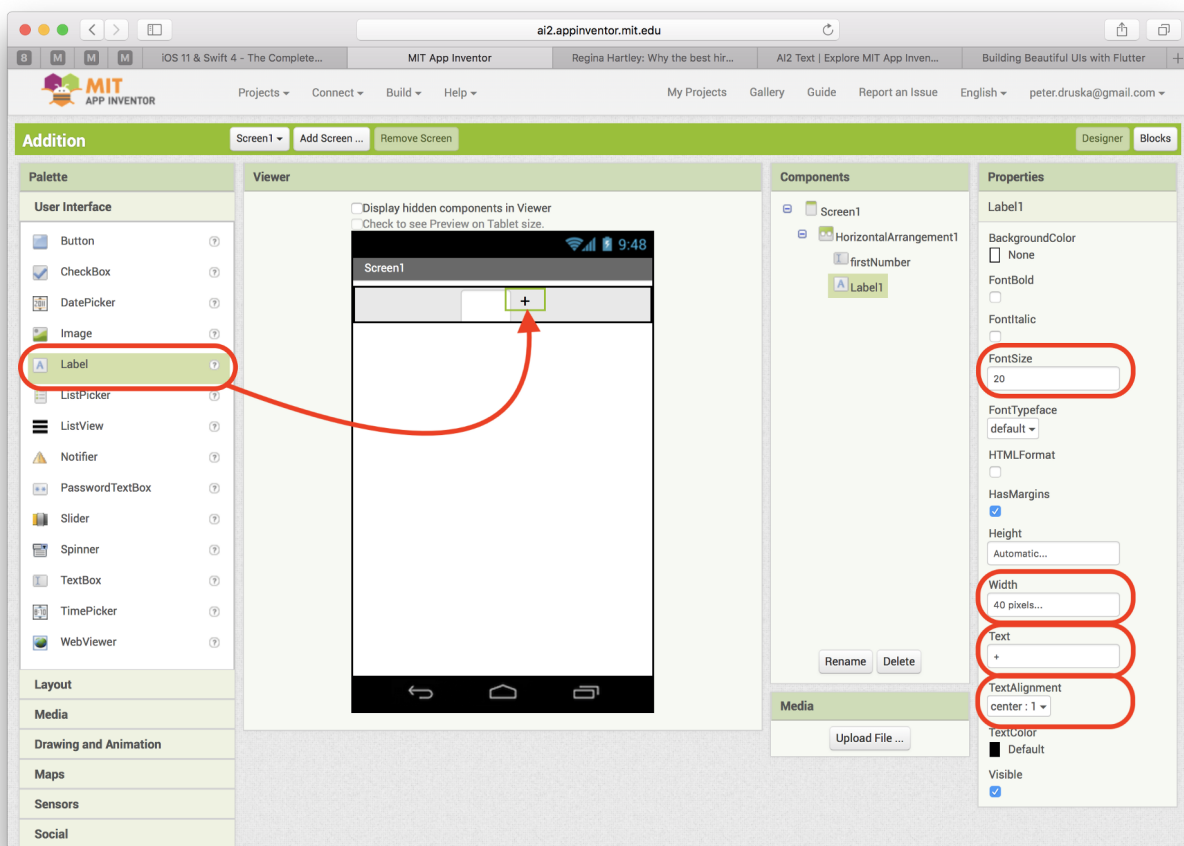
Novopridaný a nastavený textbox musíme premenovať, aby sa s ním lepšie pracovalo pri programovaní. Označíme ho v časti **Components** a klikneme na **Rename**. Otvorí sa dialógové okno, kde zapíšeme jeho meno. Ja uvádzam **firstNumber** (ang. prvé číslo). Potom klikneme na OK a je to!



**Obrázok 4.** Premenovanie elementu.

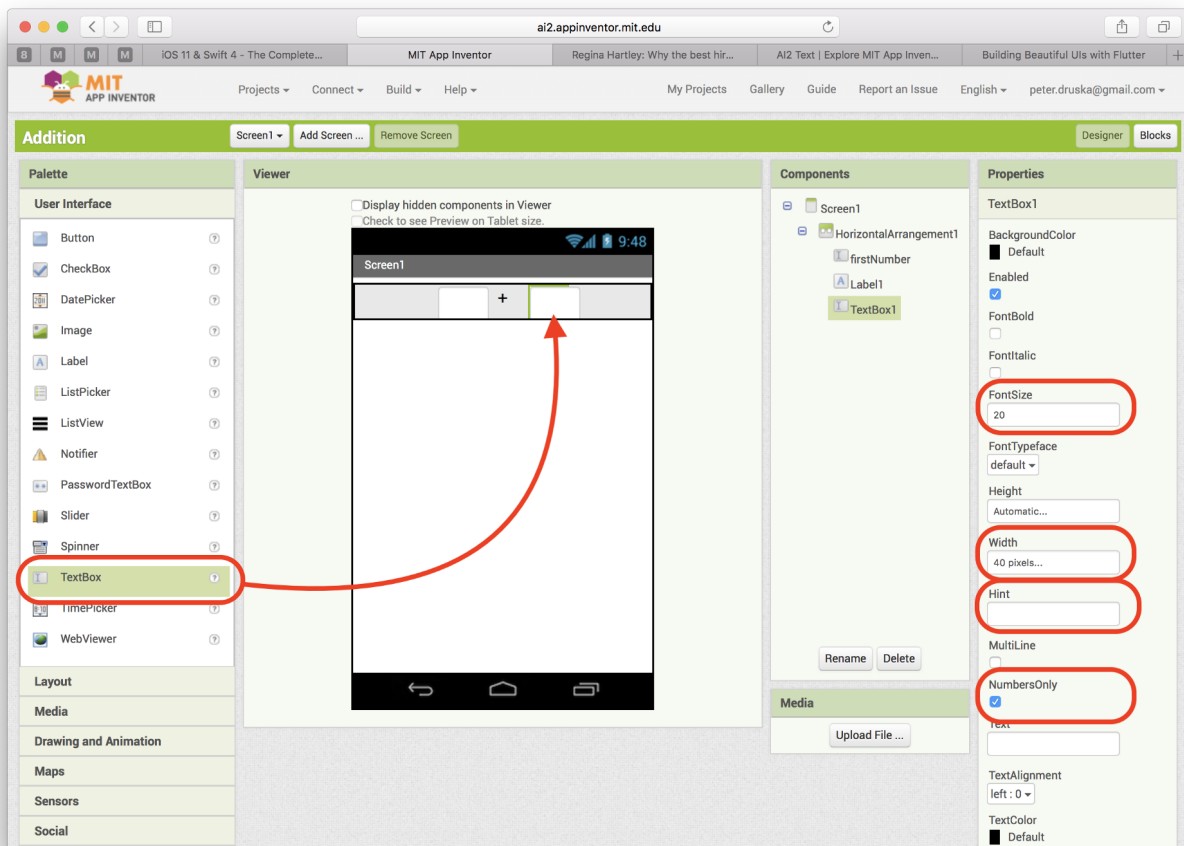
Hneď za textbox **firstNumber** pridáme textový popisok typu **Label**, ktorému prenavstavíme tieto vlastnosti:

- FontSize: 20
- Width: 40 pixels
- Text: +
- TextAlignment: center



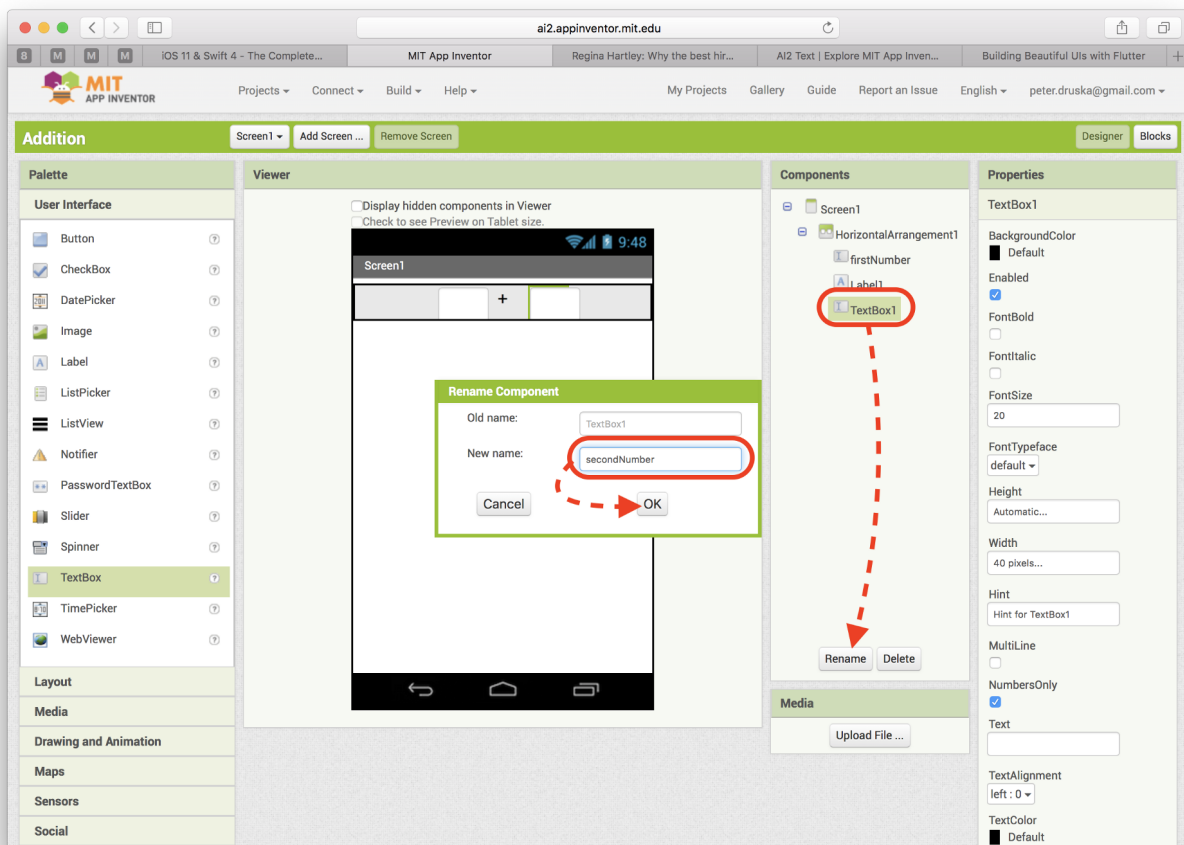
**Obrázok 5.** Pridanie textového popisku.

Pridáme nový textbox za symbol pluska (+) a nastavíme mu presne tie isté vlastnosti, ako textboxu **firstNumber**. Vid' obrázok 3.



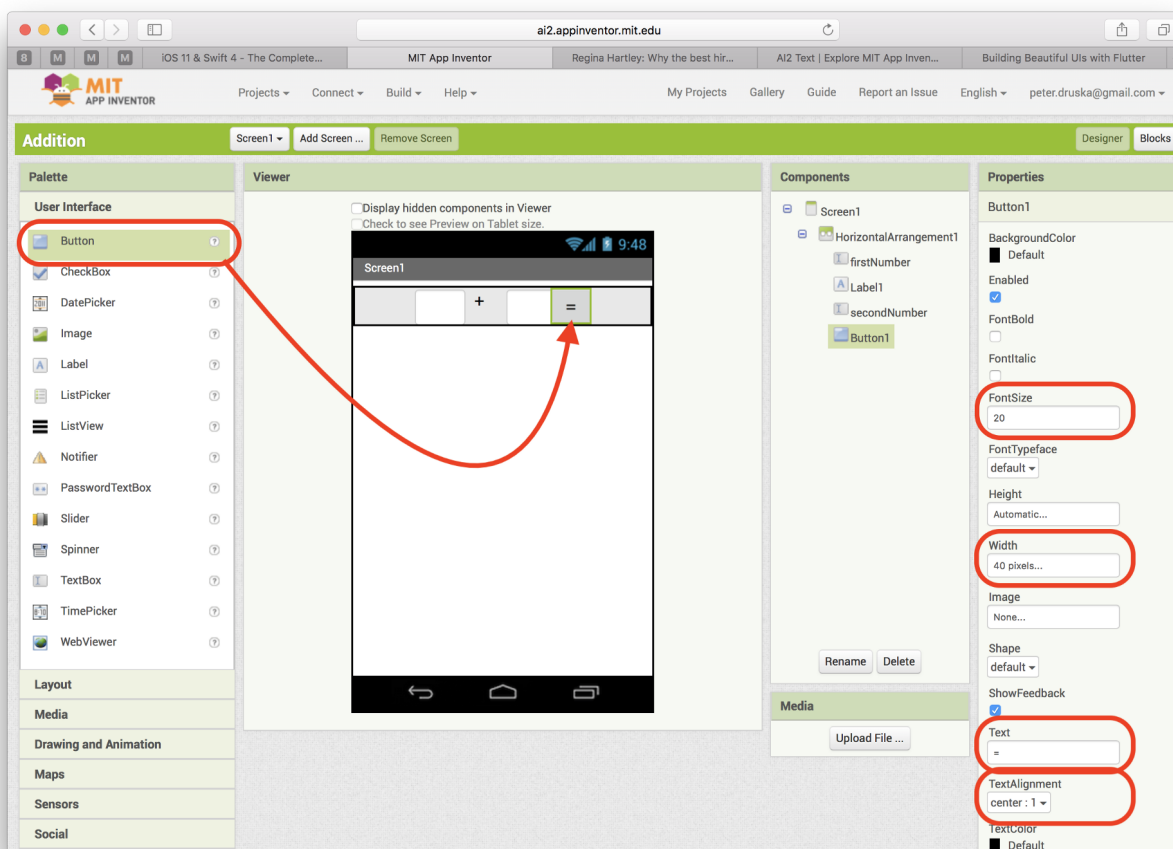
Obrázok 6. Pridanie nového textboxu.

Teraz premenujeme novopridaný textbox na **secondNumber** (ang. druhé číslo).



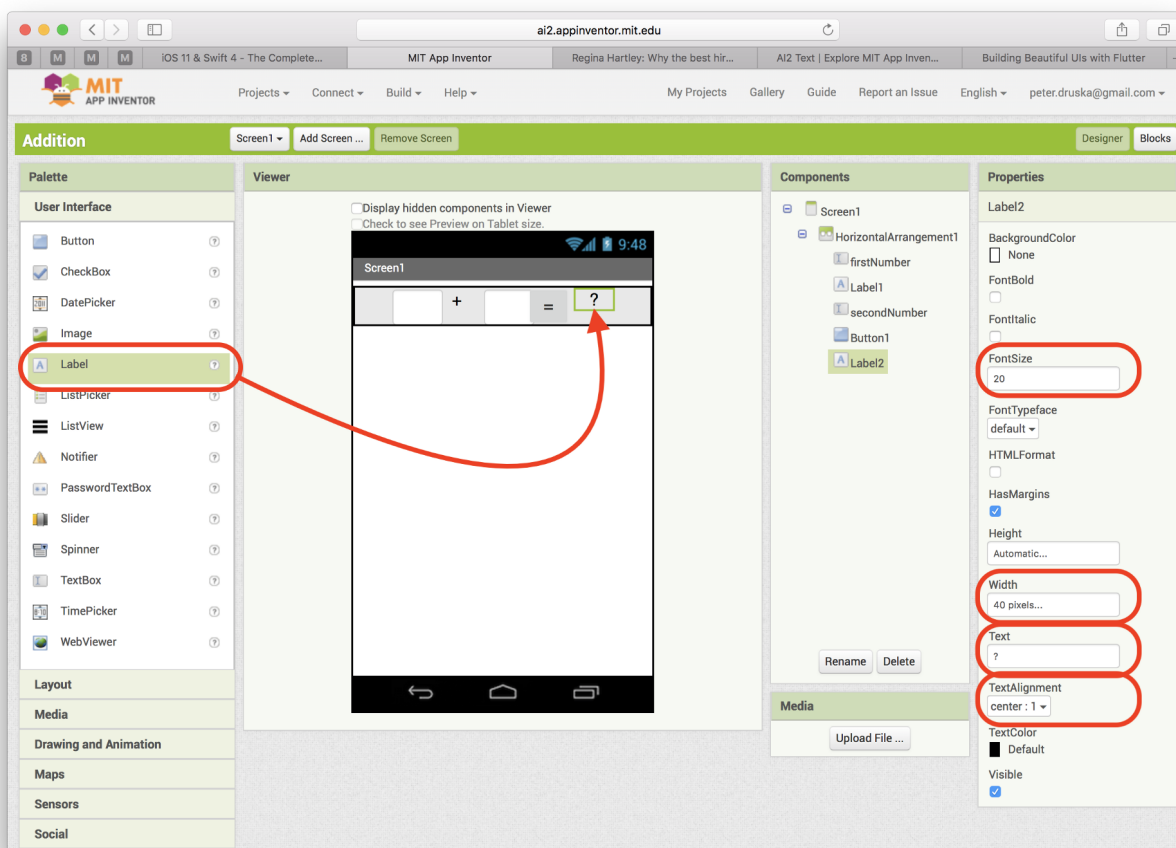
**Obrázok 7.** Premenovanie druhého textboxu.

Pridáme nové tlačidlo, ktoré bude zobrazovať znamienko *rovná sa* (=), a ktoré bude zabezpečovať výpočet, ktorého výsledok sa zobrazí napravo od neho. Nastavíme mu aj vlastnosti, ktoré vidno na obrázku 8.



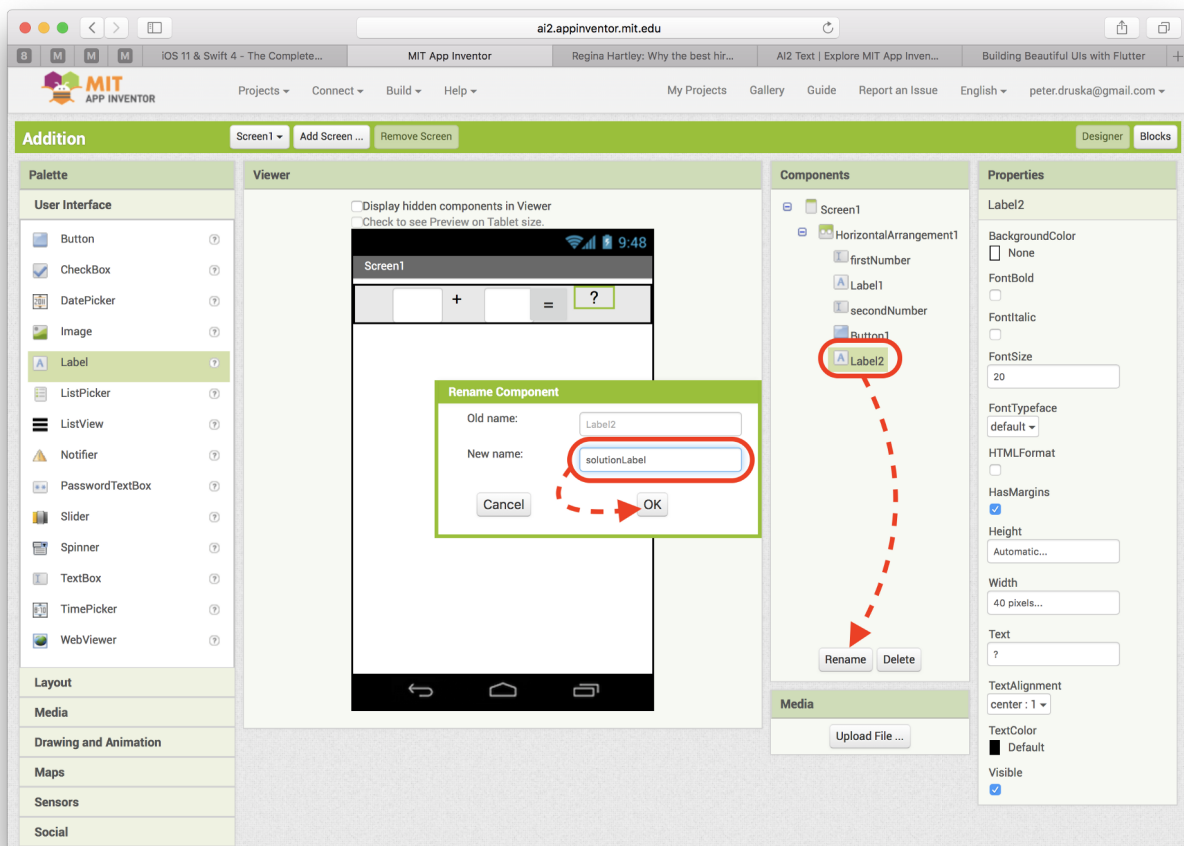
**Obrázok 8.** Pridanie tlačidla =.

Teraz pridáme nový textový popisok typu **Label** napravo od tlačidla *rovná sa*. Nastavíme mu vlastnosti tak, aby vizuálne všetko ladilo s ostatnými už pridanými prvkami. Vid' obrázok 9.



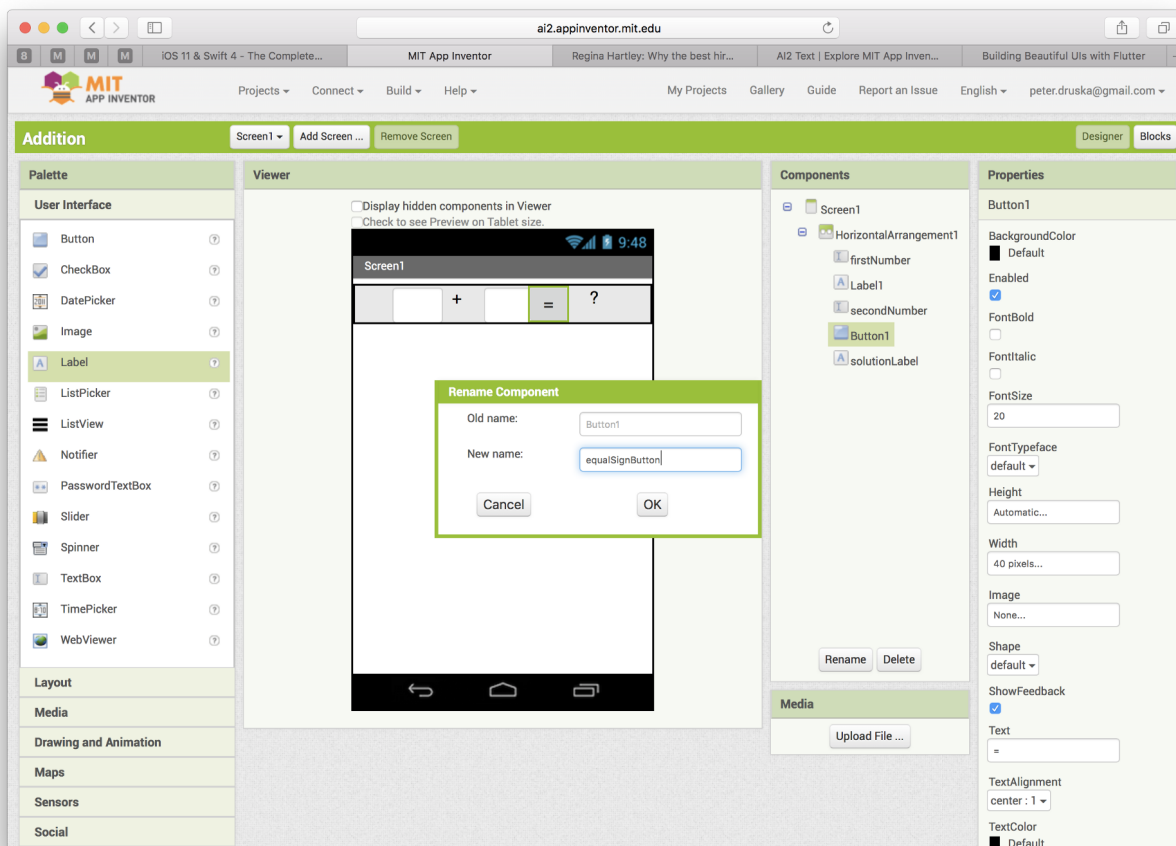
**Obrázok 9.** Pridanie textového popisku na zobrazenie výsledku.

Textový popisok **Label1** musíme premenovať tak, aby názov dával zmysel. V ukážke to je **solutionLabel**.



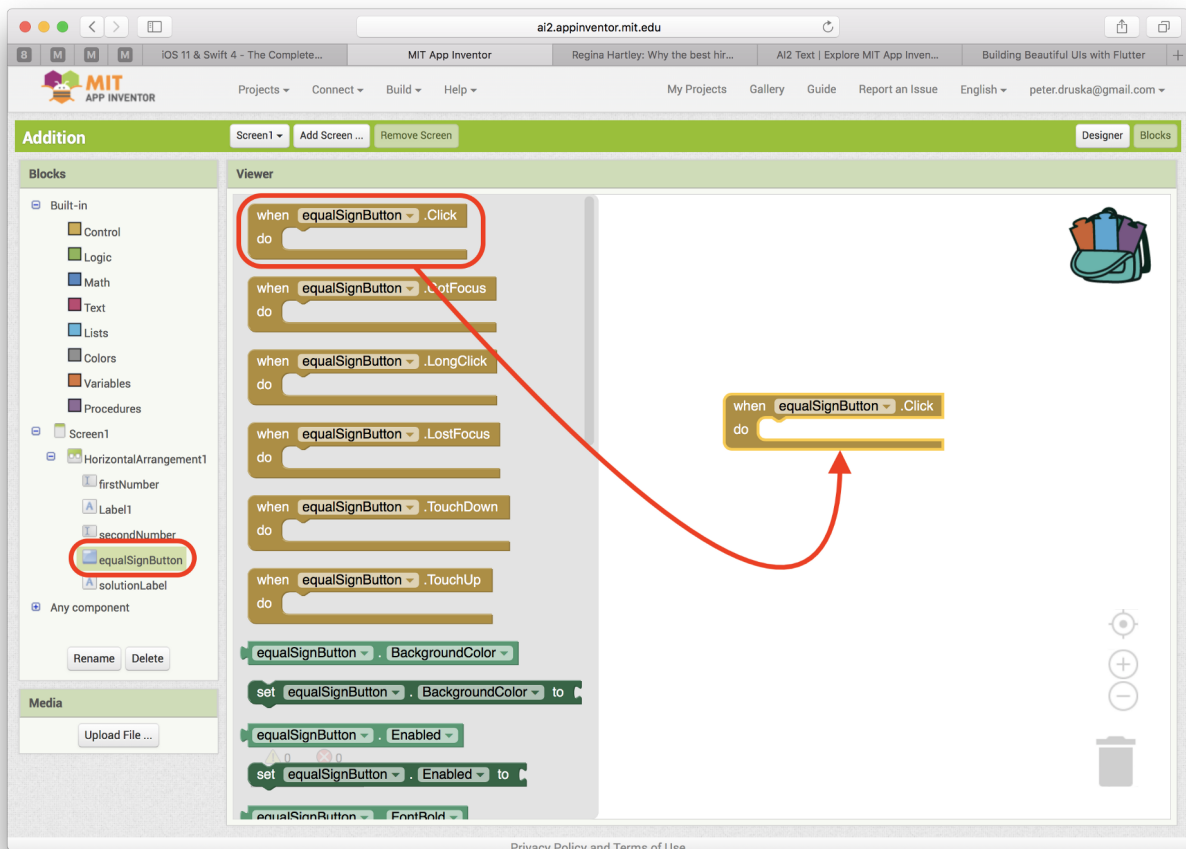
**Obrázok 10.** Premenovanie textového popisku na zobrazenie výsledku.

Premenovať by sme mali aj tlačidlo **Button1**. V ukážke uvádzame názov **equalSignButton**.



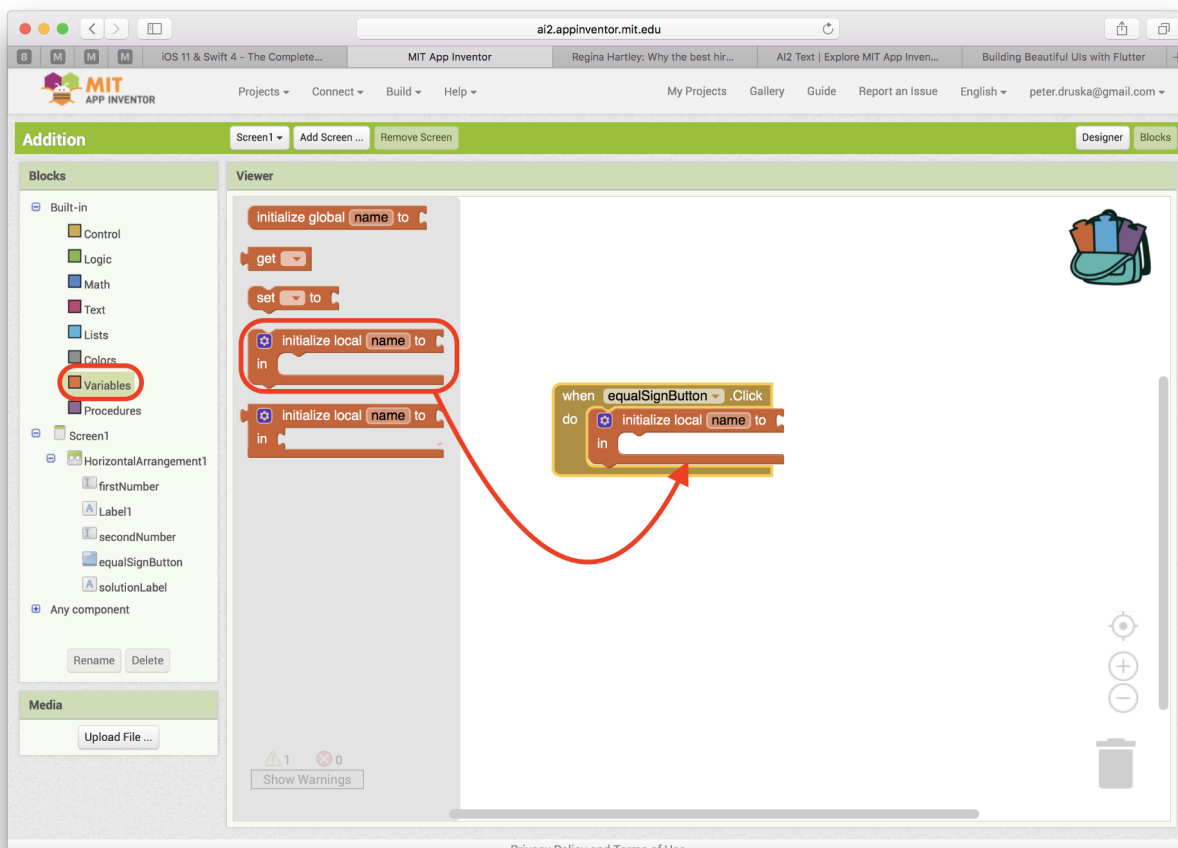
Obrázok 11. Premenovanie tlačidla =.

Teraz sa prepneme do blokov, kde naprogramujeme jediné tlačidlo **equalSignButton** tak, aby vypísal súčet dvoch zadaných čísel. Preto ho označíme a do hlavnej plochy potiahneme udalosť **when equalSignButton.Click** do.



**Obrázok 12.** Udalosť kliknutia/ŕapnutia na tlačidlo.

Keď na tlačidlo používateľ klikne/ťapne, inicializuje sa lokálna premenná.



Obrázok 13. Inicializácia lokálnej premennej.

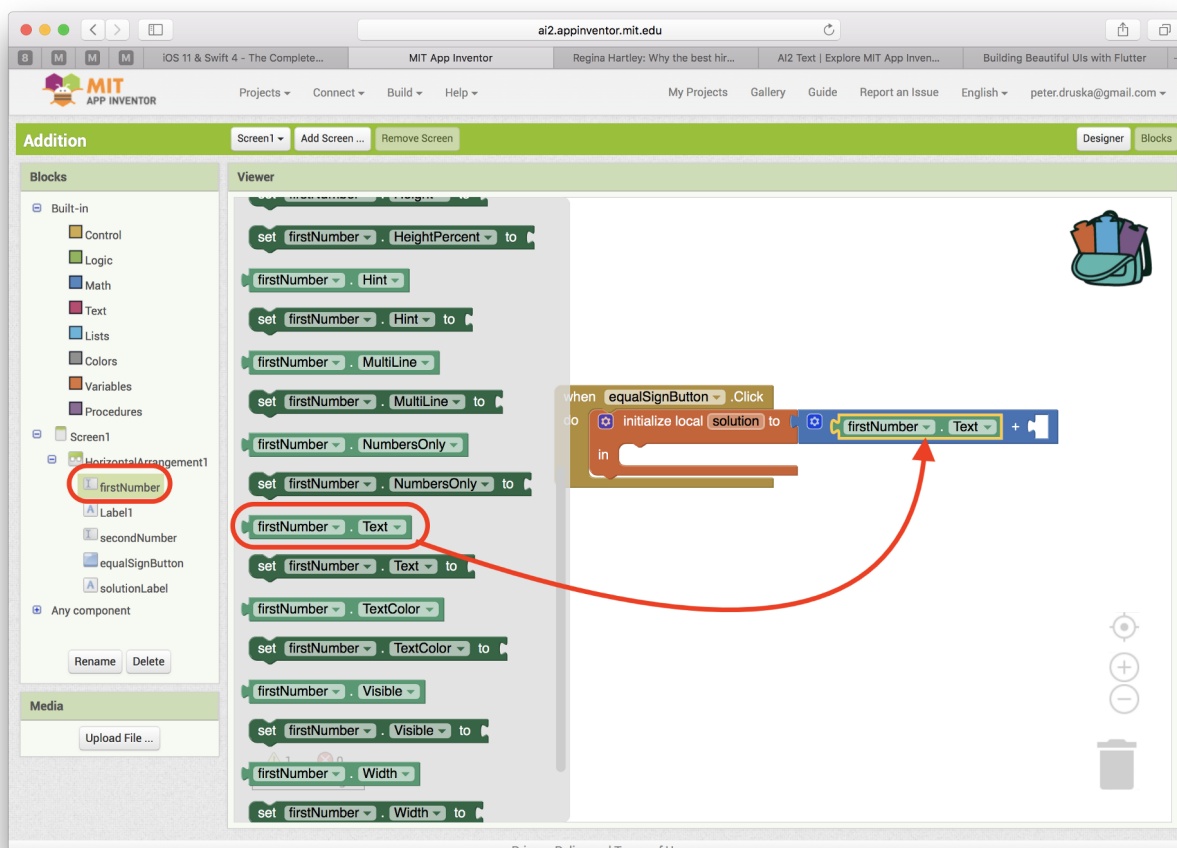
Lokálnu premennú nazveme **solution** (ang. riešenie).



Obrázok 14. Definovanie mena lokálnej premennej.

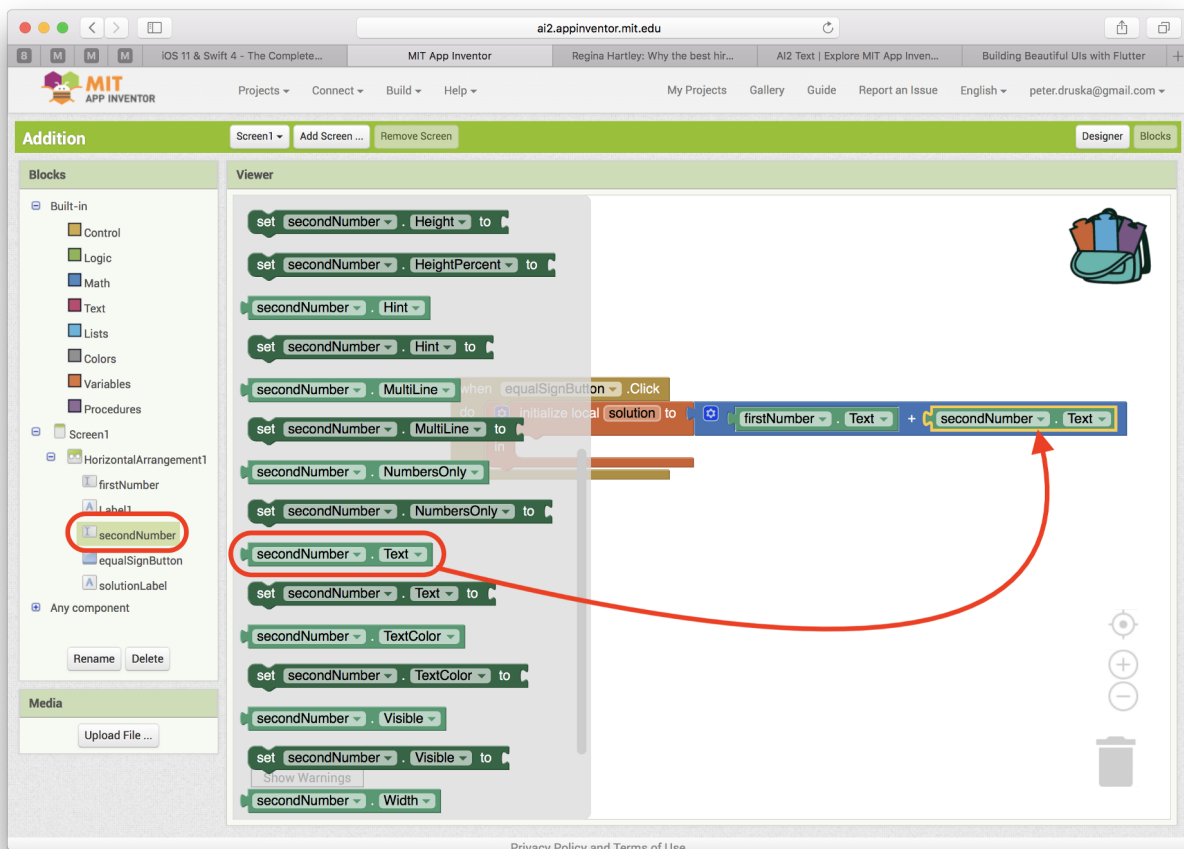


Prvé číslo (prvý operand) v súčte bude textová hodnota textboxu **firstNumber**. Môžeme to urobiť preto, lebo tento textbox sme nastavili tak, že zadávať doň sa môžu len čísla a nič iné. Vid' obrázok 3.



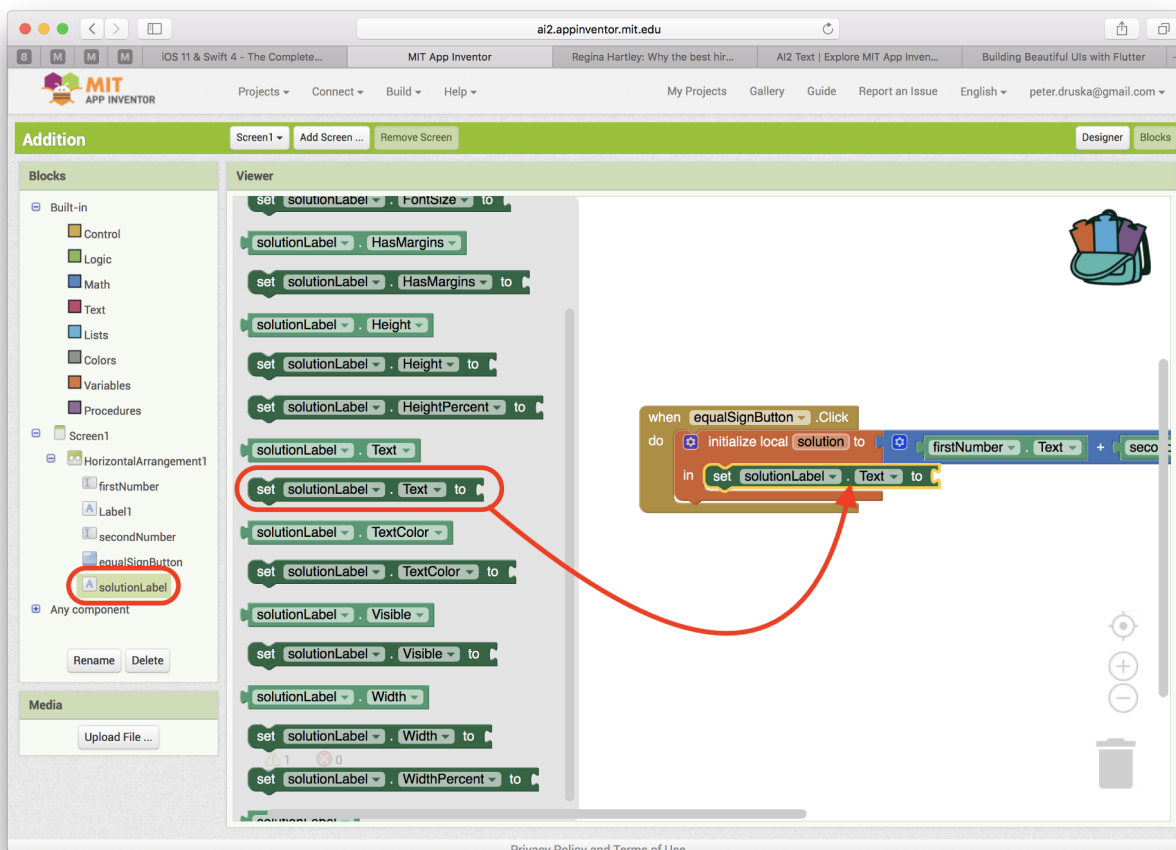
Obrázok 16. Prvý operand súčtu.

Druhé číslo (druhý operand) je textová hodnota textboxu **secondNumber**.



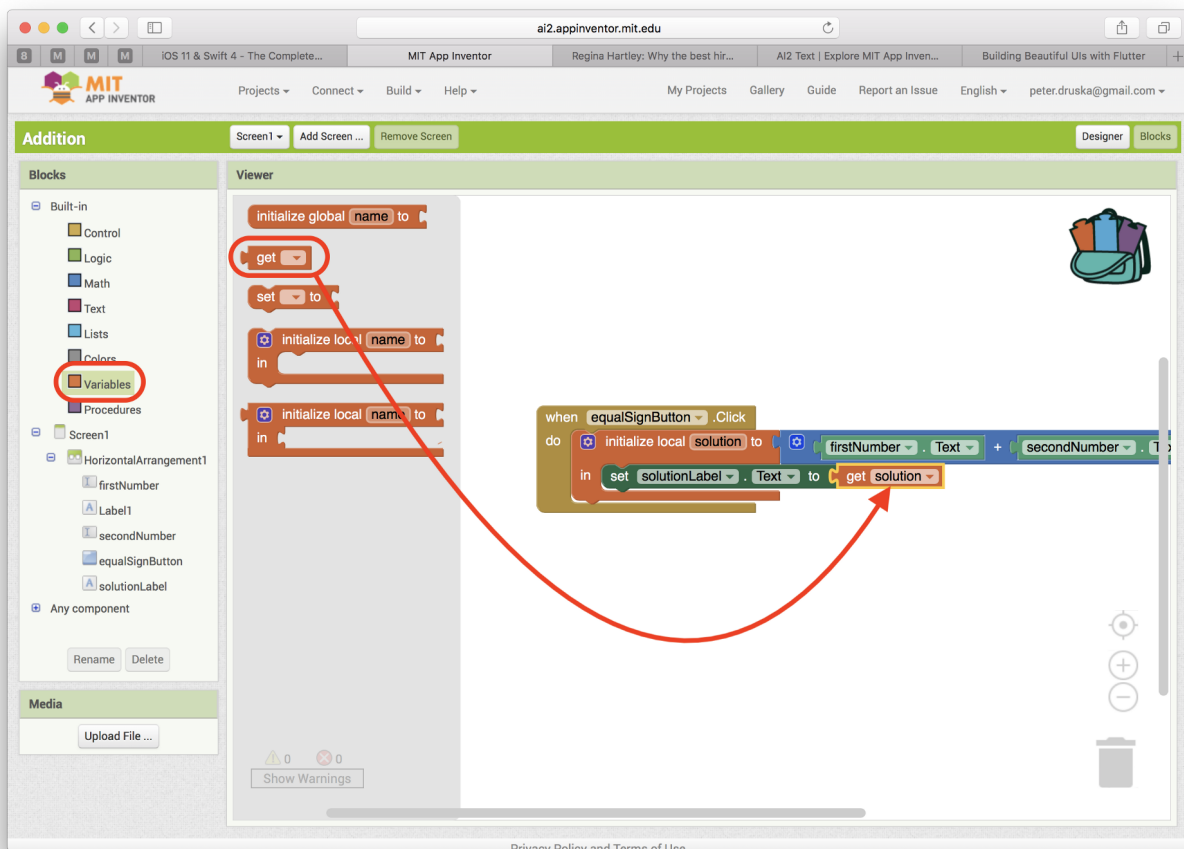
Obrázok 17. Druhý operand súčtu.

Ostáva nastaviť výsledok výpočtu textovému popisku **solutionLabel**. Zatiaľ nie je nastavený na žiadnu hodnotu.



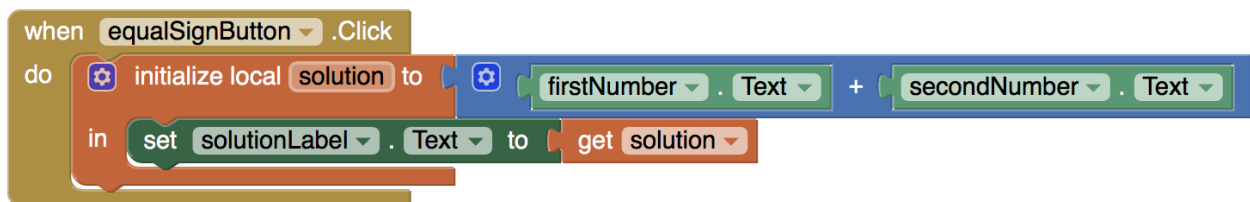
Obrázok 18. Nastavenie textového popisku.

Aby sme výsledok vypísali v popisku **solutionLabel**, musíme do jeho vlastnosti **text** pridať hodnotu premennej **solution**, kde je výsledok výpočtu uložený.



**Obrázok 19.** Volanie premennej s výslednou hodnotou.

A tu je výsledný program.



**Obrázok 20.** Výsledný program.

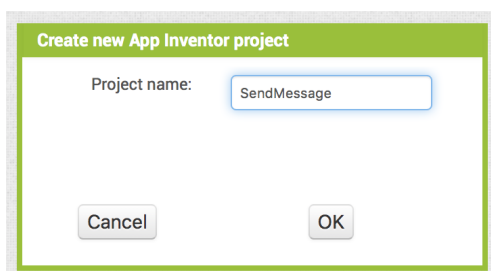
Appku zbuilduj a spusti na svojom zariadení. Všetko by malo fungovať. Neošetrili sme stavy, keď sú textboxy prázdne a je pravdepodobné, že appka v takom prípade po stlačení tlačidla + spadne.

Naším cieľom nebolo urobiť dokonalé riešenie, ale ukázať si, ako sa v princípe vytvára appka na tento účel v ApplInventore.

# Appka: odošli správu

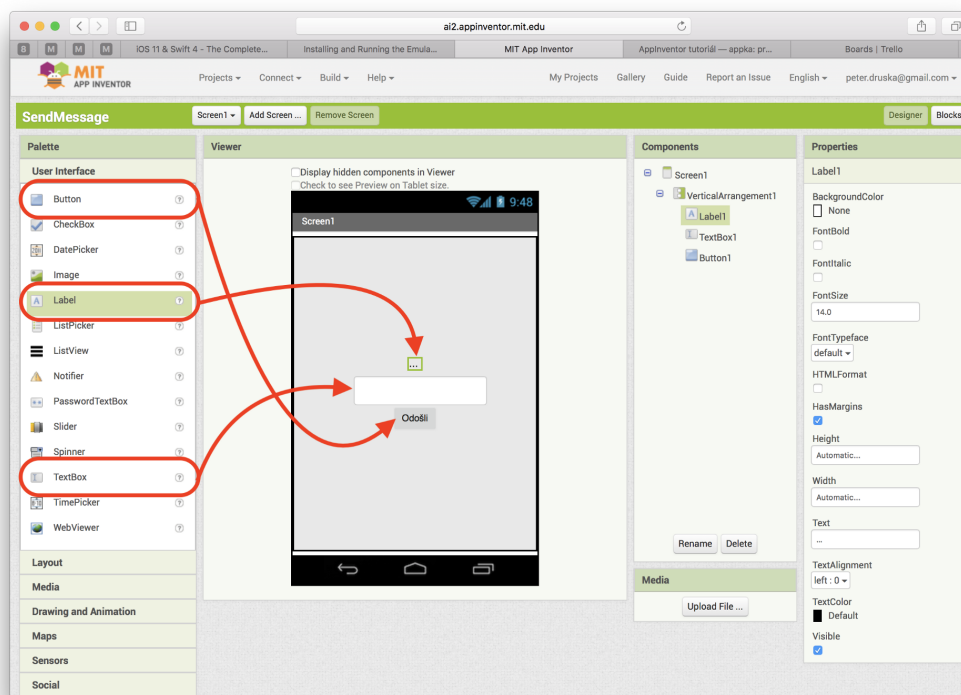
Veľmi jednoduchý návod, ako zobraziť text zadaný do textového políčka.

Ako prvé nazveme projekt nejak rozumne. V tomto prípade idem poselať správu, tak sa projekt bude volať **SendMessage**.



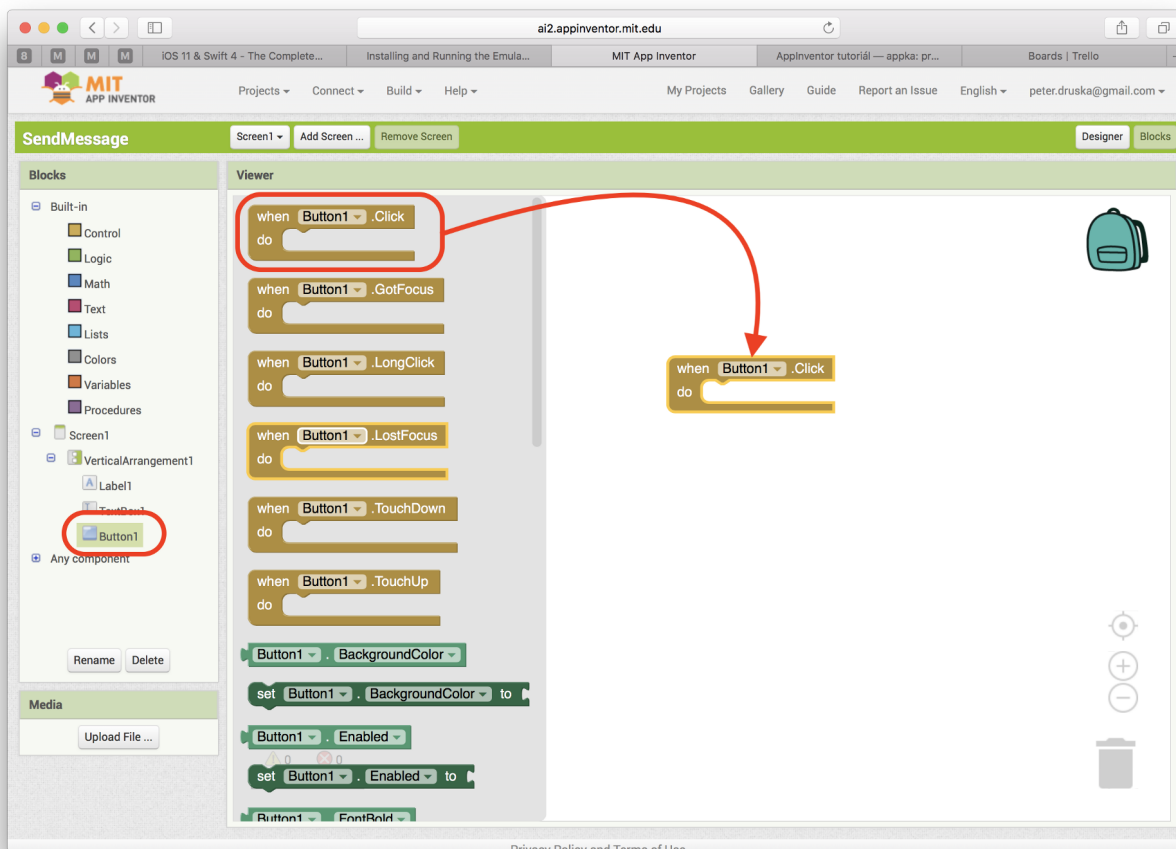
Obrázok 1. Názov projektu.

Do obrazovky aplikácie potiahneme tri prvky: **Label**, **TextBox** a **Button**. Ja ich mám navyše zavreté ešte v prvku **VerticalArrangement**, čo ale nemusíš robiť. V ukážke to je na to, aby to rozhranie vyzeralo dobre usporiadané. Nevadí, keď to v tomto prípade budeš mať inak usporiadané.



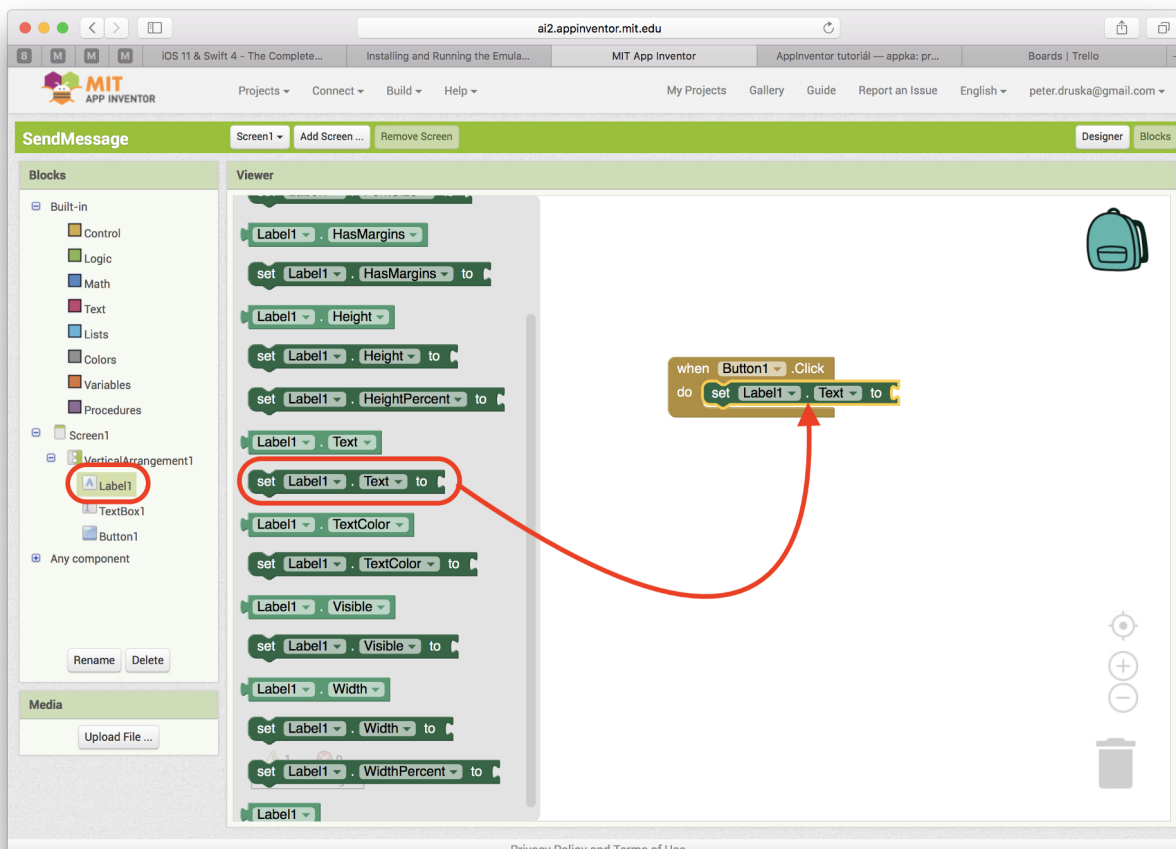
Obrázok 2. Funkčné prvky v obrazovke.

Prepnime sa do blokov a pre tlačidlo **Button1** vytvoríme udalosť **when Button1.Click** do.



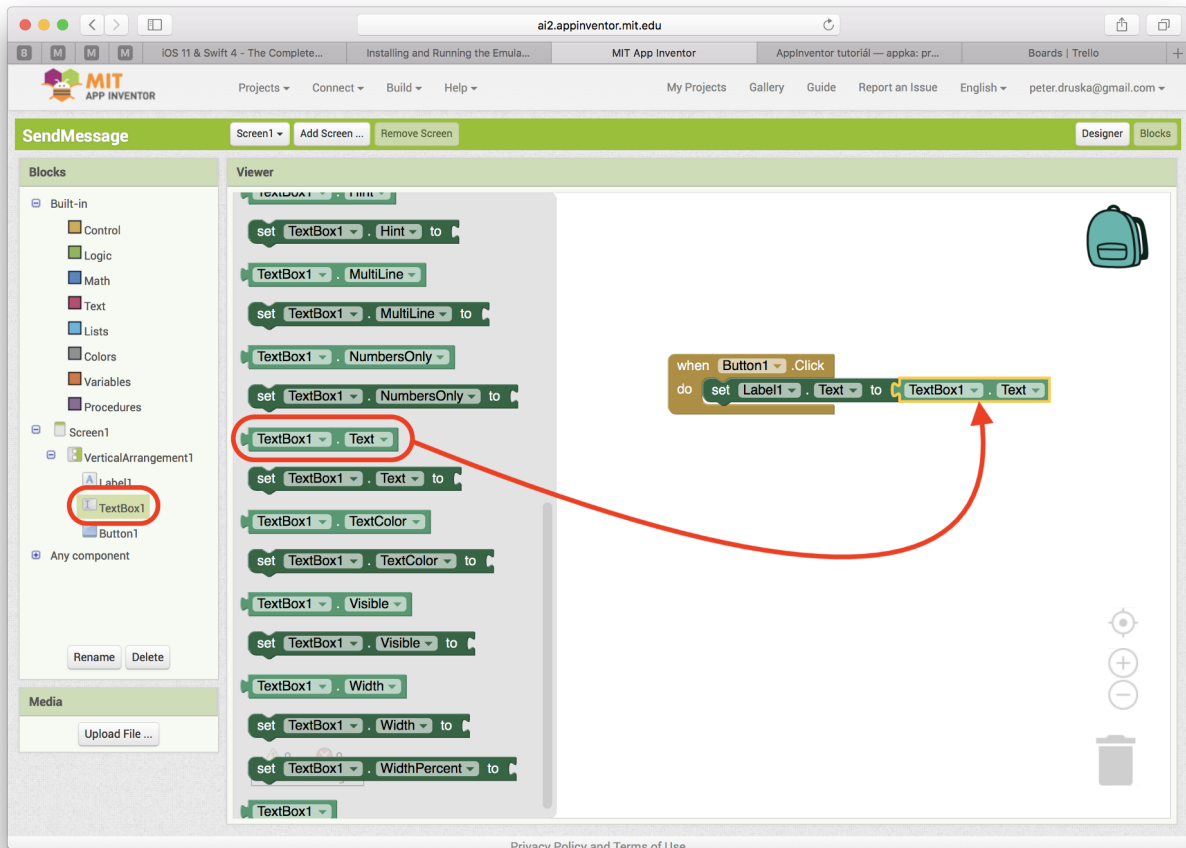
Obrázok 3. Udalosť kliknutia pre tlačidlo.

Po kliknutí chceme, aby sa nastavil textovému popisku **Label1** nejaký text. Jedná sa o blok `set Label1.Text to`.



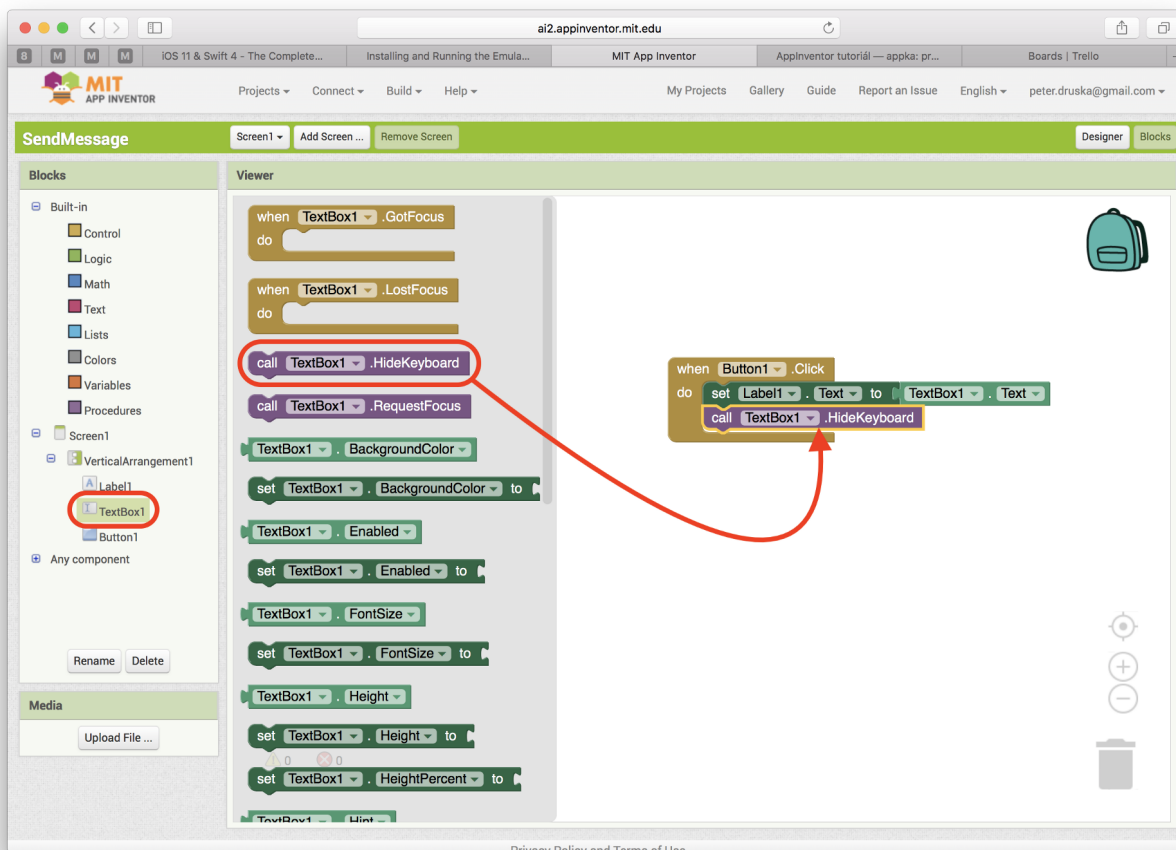
Obrázok 4. Blok nastavenia textu textovému popisku.

Predposledným krokom je vybrať text z políčka na zadanie textu **TextBox1** blokom **TextBox1.Text**. Čokoľvek doň napíšeme, to sa zobrazí v textovom popisku **Label1** po stlačení tlačidla **Button1**.



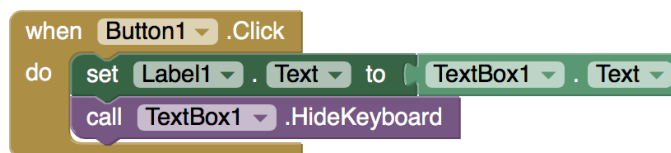
**Obrázok 5.** Pridanie textu do popisku z políčka na jeho zadanie.

Ako posledný krok pridáme blok textu `call TextBox1.HideKeyboard`, ktorý zabezpečí, že po stlačení tlačidla a zobrazení textovej správy sa skryje klávesnica.



Obrázok 6. Skrývanie klávesnice.

Tadá! Tu je celý blok programu. Vyskúšaj appku na svojom zariadení.



Obrázok 7. Celý program.

Do veselého kódovania!

## Záver tejto príručky je len začiatok

Tento učebný materiál je len začiatok. Pre viac tutoriálov na programovanie aplikácií v rozhraní App Inventor navštív web [www.medium.com](http://www.medium.com), kde vyhľadaj kľúčové slová: „App Inventor Tutorial Harpúna“. Neustále dopĺňame nové, preto ich nie je možné dopíňať do tohto papierového materiálu.

Veľmi veľa tutoriálov sa dá nájsť aj na oficiálnom webe <http://appinventor.mit.edu/>, prípadne na [www.youtube.com](http://www.youtube.com).