

Project number: 2023-1-PL01-KA220-SCH-000154043

**IoT4Schools**

**“Bringing the Internet of Things in school education as a tool to address 21<sup>st</sup> century challenges”**

## **Monitorowanie aktywności fizycznej: tworzenie krokomierza**

**Przewodnik dla nauczycieli**

**Autorzy: C.Papasarantou, R. Alimisi**

**Organizacja: EDUMOTIVA**

*License: CC BY-NC 4.0 LEGAL CODE, Attribution-NonCommercial 4.0 International*



**Co-funded by  
the European Union**

*The European Commission's support to produce this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.*

## Spis treści

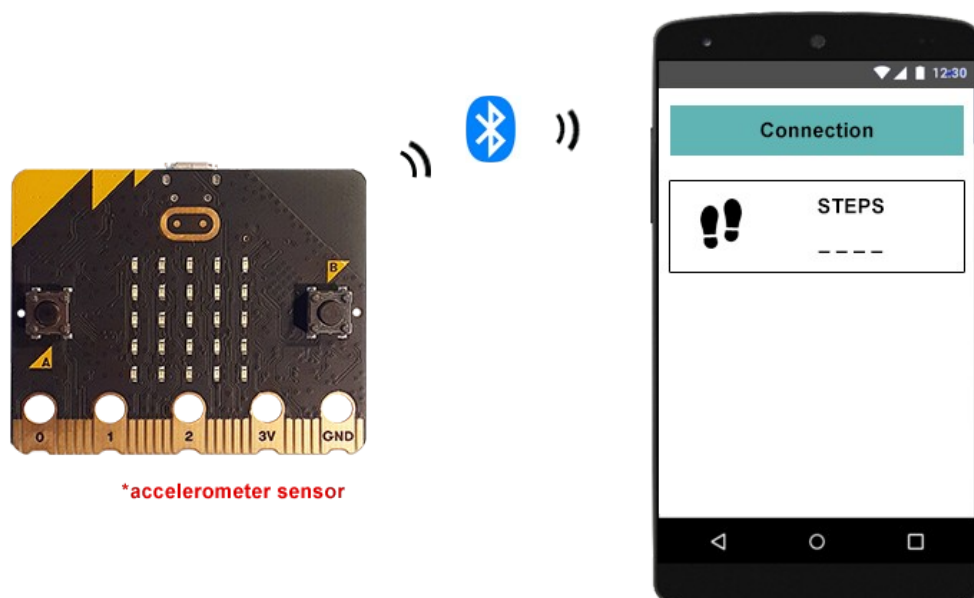
1 Wprowadzenie do projektu.....	3
1 Wprowadzenie do projektu.....	3
1.1 Ogólny scenariusz i zakres projektu.....	3
1.1 Ogólny scenariusz i zakres projektu.....	3
1.2 Cele uczenia się.....	4
1.2 Cele uczenia się.....	4
1.3 Ścieżka uczenia się – Etapy wprowadzania projektu.....	5
1.3 Ścieżka uczenia się – Etapy wprowadzania projektu.....	5
1.4 Wymagania wstępne.....	5
1.4 Wymagania wstępne.....	5
1.5 Wymagany sprzęt i oprogramowanie.....	5
1.5 Wymagany sprzęt i oprogramowanie.....	5
1.6 Plan czasowy.....	6
1.6 Plan czasowy.....	6
2 Wdrażanie projektu.....	6
2 Wdrażanie projektu.....	6
2.1 Poziom 1: Tworzenie krokomierza liczącego kroki.....	6
2.1 Poziom 1: Tworzenie krokomierza liczącego kroki.....	6
2.1.1 Proces tworzenia układu.....	6
2.1.1 Proces tworzenia układu.....	6
2.1.2 Programowanie.....	6
2.1.2 Programowanie.....	6
2.1.3 Crafting.....	18
2.1.3 Crafting.....	18
2.2 Poziom 2: Programowanie krokomierza do rejestrowania wielu danych.....	19
2.2 Poziom 2: Programowanie krokomierza do rejestrowania wielu danych.....	19
2.2.1 Mierzenie przebytej odległości z wykorzystaniem krokomierza.....	19
2.2.1 Mierzenie przebytej odległości z wykorzystaniem krokomierza.....	19
2.2.2 Programowanie.....	19
2.2.2 Programowanie.....	19
3 Wskazówki i zalecenia.....	21
3 Wskazówki i zalecenia.....	21

3.1 Dalszy rozwój projektu.....	21
3.1 Dalszy rozwój projektu.....	21
3.2 Personalizacja krokomierza.....	21
3.2 Personalizacja krokomierza.....	21
3.3 Sprawdzenie czy micro:bit można połączyć poprzez Bluetooth.....	21
3.3 Sprawdzenie czy micro:bit można połączyć poprzez Bluetooth.....	21
3.4 Omówienie zalet i wad.....	21
3.4 Omówienie zalet i wad.....	21
4 Materiał źródłowy.....	21
4 Materiał źródłowy.....	21

## 1 Wprowadzenie do projektu

### 1.1 Ogólny scenariusz i zakres projektu

Celem projektu jest zapoznanie uczniów z koncepcją IoT w kontekście monitorowania stanu zdrowia poprzez aktywność fizyczną. Uczniowie nauczą się jak stworzyć swój własny krokomierz (np. fitness tracker- urządzenie służące do pomiaru aktywności fizycznej, wykrywające ruch oraz liczbę kroków , tym samym otrzymując przybliżony zakres przebytego dystansu) oraz jak udoskonalić aplikację, która otrzymuje dane z krokomierza i wyświetla je w czasie rzeczywistym. W szczególności używając procesora the BBC micro:bit i wbudowanego czujnika prędkości, uczniowie nauczą się jak stworzyć i zaprogramować swój własny krokomierz, który może monitorować aktywność fizyczną poprzez liczenie kroków i przebyty dystans. Używając oprogramowania the MIT App Inventor (<https://appinventor.mit.edu/>) będą uczyć się jak zaprojektować i zaprogramować aplikację, która poda i wyświetli policzone kroki. Dane będą przekazywane przez Bluetooth. Ogólny zamysł projektu jest przedstawiony na ilustracji 1. Ten projekt pomoże uczniom zrozumieć jak różne urządzenia mogą wymieniać się danymi zdalnie, równocześnie zapoznając ich z procesem monitorowania danych w czasie rzeczywistym i podejmowaniu dalszych decyzji w oparciu o otrzymane dane.



*Rysunek 1: Graficzna reprezentacja koncepcji projektu*

Umieszczenie tego projektu w kontekście edukacji szkolnej powoduje, że wesprze on zarówno uczniów jak i nauczycieli w doskonaleniu ich umiejętności i kompetencji technologicznych. Ponadto, projekt wspiera interdyscyplinarność oraz integrację z głównymi założeniami STEM, poprzez realizację praktycznych działań pochodzących z różnych dziedzin naukowych, włączając technologię i matematykę. Dodatkowo, materiały edukacyjne (przewodnik nauczyciela; karty pracy uczniów etc.) w sposób efektywny wesprą edukatorów w procesie wdrażania tego projektu w klasie.

## 1.2 Cele uczenia się

Dzięki temu projektowi uczniowie będą mogli:

- Zbudować i zaprogramować urządzenie, które liczy kroki;
- Zaprojektować i zaprogramować aplikację, która wyświetla liczbę kroków;
- Zaprogramować urządzenie do mierzenia dystansu opartego na policzonych już krokach;
- Nauczyć się jak korzystać z czujników takich jak akcelerometr-przyspieszeniometer, aby monitorować dane odnoszące się do stanu zdrowia ( biorąc pod uwagę aktywność fizyczną);
- Zrozumieć jak urządzenia IoT mogą zbierać i przekazywać dane przez Bluetooth;
- Zrozumieć i wyjaśnić jak dane mogą być monitorowane w czasie rzeczywistym;
- Rozpoznawać zalety i wady oraz ryzyko wynikające z korzystania z takich urządzeń i aplikacji, w odniesieniu do podejmowania decyzji związanych ze stanem zdrowia;

### 1.3 Ścieżka uczenia się – Etapy wprowadzania projektu

Projekt dotyczy kwestii zdrowia, a przede wszystkim sposobów, które dzięki IoT mogą być wdrożone w celu poprawy ogólnego stanu zdrowia, poprzez monitorowanie danych, odnoszących się do codziennej aktywności fizycznej. Właśnie w tym celu, uczniowie będą zachęceni do stworzenia swojego własnego krokomierza, poprzez użycie micro:bit board.

Sugerowane etapy poszczególnych działań realizacji projektu z uczniami:

**Tworzenie grup:** podzielić uczniów na 2- lub 3-osobowe grupy;

**Burza mózgów:** zachęcić uczniów do wyszukiwania informacji, dotyczącej krokomierza (jak działa; jakie dane monitoruje; dokąd są wysyłane i gdzie są przechowywane; w jaki sposób mogą być wykorzystane, żeby kontrolować aktywności fizyczne). Jeśli uczniowie korzystali już z podobnego urządzenia, niech podzielą się swoim doświadczeniem (jakie dane przeważnie sprawdzali; jak często monitorowali postępy w podejmowanej aktywności fizycznej; jakie wnioski nasunęły się w oparciu o pozyskane dane etc.)

**Dyskusja nad podjętymi zadaniami:** każda grupa powinna podzielić się informacjami, które zdobyła i omówić najbardziej istotne aspekty wykorzystania krokomierza. Należy podkreślić szczególny cel projektu, mianowicie: stworzenie urządzenia, które liczy kroki oraz opracowanie aplikacji, która wyświetla policzone kroki.

**Planowanie:** każda grupa powinna przemyśleć kwestię konstrukcji urządzenia (wygodny sposób umieszczenia krokomierza tzn. przy której części ciała) oraz zaprojektowania aplikacji (wygląd).

**Tworzenie:** korzystając z kart pracy, każda grupa tworzy swój własny krokomierz oraz własną aplikację do wyświetlania policzonych kroków.

**Testowanie – optymalizacja:** po wykonaniu projektu, uczniowie testują stworzone przez siebie urządzenia. Można zasugerować, żeby grupy wymieniły się urządzeniami, w celu sprawdzenia czy są jakieś różnice w sposobie działania.

**Prezentacja prac projektowych:** uczniowie prezentują efekty swojej pracy na forum, dzieląc się doświadczeniami, jakie zdobyli podczas pracy nad urządzeniami oraz dyskutują o wpływie takich urządzeń na codzienne aktywności fizyczne, wskazując plusy i minusy korzystania z krokomierza, w celu ich monitorowania.

### 1.4 Wymagania wstępne

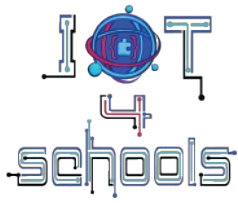
Uczniowie powinni znać podstawowe metody programowania blokowego.

### 1.5 Wymagany sprzęt i oprogramowanie

Sprzęt:

- płytki BBC micro:bit
- Zewnętrzne zasilanie: pojemnik na 2 baterie AAA bądź power bank z niskim napięciem wyjściowym (do 5V)
- Czujnik tętna (opcjonalny/ tylko do poziomu 2)

Oprogramowanie:



- środowisko Microsoft Makecode
- MIT App Inventor

## 1.6 Plan czasowy

Czas potrzebny do wykonania projektu: 4- 6 godzin.

Sugerowany podział przeznaczonego czasu:

- 30-40 minut – omówienie projektu (łącznie z burzą mózgów i dyskusją)
- 30-40 minut - planowanie
- 1-2 godziny – ukończenie poziomu 1
- 1 godzina – ukończenie poziomu 2
- 30 minut – podsumowanie i dyskusja

## 2 Wdrażanie projektu

### 2.1 Poziom 1: Tworzenie krokomierza liczącego kroki

#### 2.1.1 Proces tworzenia układu

*Na potrzeby tego projektu zostanie wykorzystany wbudowany akcelerometr. Dlatego pod koniec projektu wystarczy podłączyć zewnętrzne źródło zasilania, takie jak uchwyt na baterie 2AAA, aby odłączyć micro:bit od komputera i przekształcić go w urządzenie przenośne.*

*Dlatego podłącz micro:bit do komputera za pomocą kabla USB i zacznij programować.*

#### 2.1.2 Programowanie

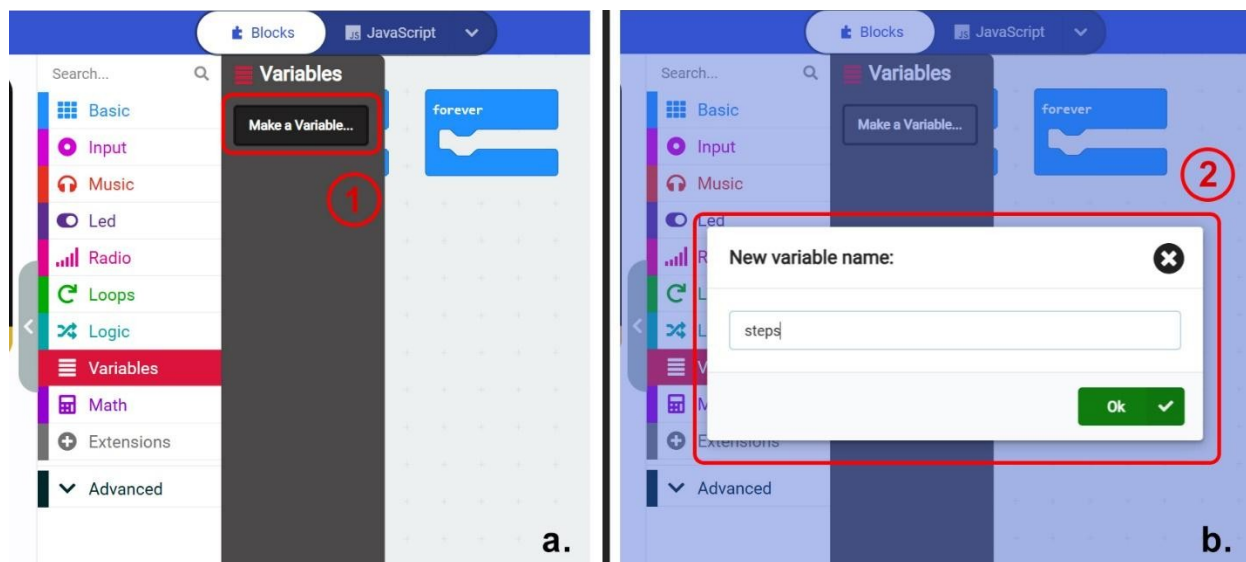
Aby zapewnić płynną realizację projektu, zaleca się rozpoczęcie od ćwiczenia próbnego, podczas którego uczniowie utworzą krokomierz liczący kroki i wyświetlający wyniki na mikro:bitach na ekranie LED.

Otwórz środowisko blokowe Microsoft Makecode (<https://makecode.microbit.org/>) i utwórz nowy projekt.

### Ćwiczenia wstępne:

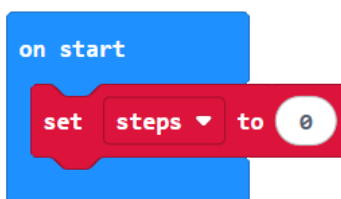
Utwórz zmienną do przechowywania zliczonych kroków. Aby to zrobić, kliknij grupę poleceń **Variables** i dalej przycisk **Make a Variable** (rysunek 2a). Następnie w wyskakującym okienku (rysunek 2b) wpisz nazwę pliku

zmienna (tj. kroki). Pod przyciskiem **Make a Variable** pojawią się dwa nowe polecenia: **set...to...** oraz polecenia **change...by...**

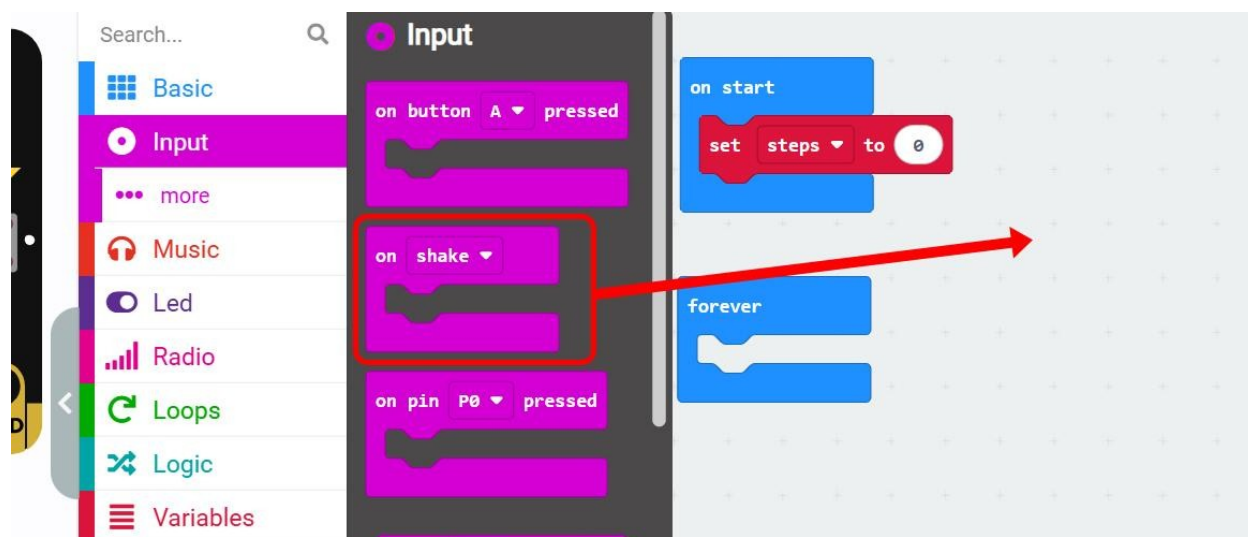


Rysunek 2: Tworzenie zmiennej

W bloku poleceń **On start** kliknij **set 'steps' to '0'**, za każdym razem gdy ustawisz licznik kroków na zero program zacznie działać.

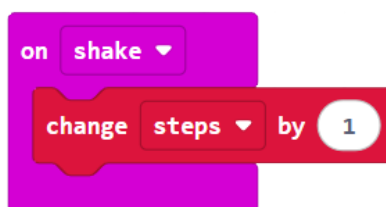


Następnie przeciągnij i upuść polecenie **On shake** z grupy poleceń **Input** do edytora skryptów (Zdjęcie 3). To polecenie aktywuje zdarzenie za każdym razem, gdy micro:bit zostanie potrząśnięty.



Rysunek 3: Znalezienie i przeciągnięcie polecenia "On shake"

W tym poleceniu kliknij polecenie **change 'step' by...** i ustaw wartość na 1. W ten sposób wartość zmienna kroku zmieni się o 1 za każdym razem, gdy mikro:bit zostanie potrząśnięty (tj. zostanie wykryty ruch poziomy lub pionowy) .



Po wykonaniu powyższych czynności, KROKOMIERZ jest prawie gotowy do użytku . Pozostało już tylko zaprogramowanie micro:bit , aby wyświetlał policzone kroki .

Z menu po prawej stronie wybierz polecenie **FOREVER** a następnie po pojawieniu się komendy **SHOW NUMBER** ustaw zmienną **STEPS** w polu wartości ( zdjęcie 4 )



*Rysunek 4: Skrypt do ćwiczenia wstępnego*

Prześlij skrypt do micro:bit i przetestuj działanie krokomierza.

### W kierunku zastosowania IoT

W ten sposób stworzyłeś przenośne urządzenie, które potrafi liczyć kroki. Dane te nie są jednak przekazywane ani przechowywane w dowolnym miejscu innym niż micro:bit. Aby móc przysyłać te dane i lepiej monitorować swoje kroki, jedynym rozwiązaniem jest stworzenie aplikacji, która w dowolnym momencie będzie odbierać zliczone kroki z micro:bit, oraz wyświetlać je na smart urządzeniu. Micro:bit posiada antenę Bluetooth, dzięki czemu może łączyć się z innymi urządzeniami Bluetooth. Dlatego należy utworzyć aplikację za pomocą MIT App Inventor, aby zainstalować ją na smart urządzeniach i wymieniać dane z micro:bit poprzez Bluetooth. Należy także dokonać zmian w istniejącym pliku skryptu micro:bit, aby włączyć połączenie Bluetooth. Z tego powodu rozwiązanie programistyczne jest podzielone na dwie części: część programistyczną Makecode i część programistyczną dla twórców aplikacji MIT.

### Część programistyczna Makecode

Aby umożliwić micro:bit wymianę danych za pomocą anteny micro:bit, należy zaimportować moduł Bluetooth Command group do swojego projektu (zobacz, jak to zrobić w pliku przewodnika technicznego micro:bit).

Skrypt, który za chwilę utworzysz, poinstruuje micro:bit, aby korzystał z dostępnych usług Bluetooth. To następuje, kiedy jest podłączony do innego urządzenia Bluetooth i przesyła dane o zliczonych krokach, gdy przycisk A jest wciśnięty.

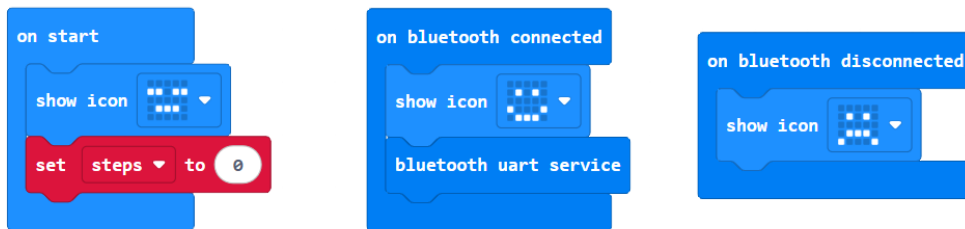
Wskazówka: Przed wprowadzeniem jakichkolwiek zmian w istniejącym kodzie/skrypcie zapisz projekt pod nową nazwą.

W istniejącym skrypcie usuń polecenie blokowania FOREVER . Następnie dodaj blok ON BLUETOOTH CONNECTED.

Polecenie z grupy poleceń Bluetooth i wybierz polecenie BLUETOOTH UART SERVICE.

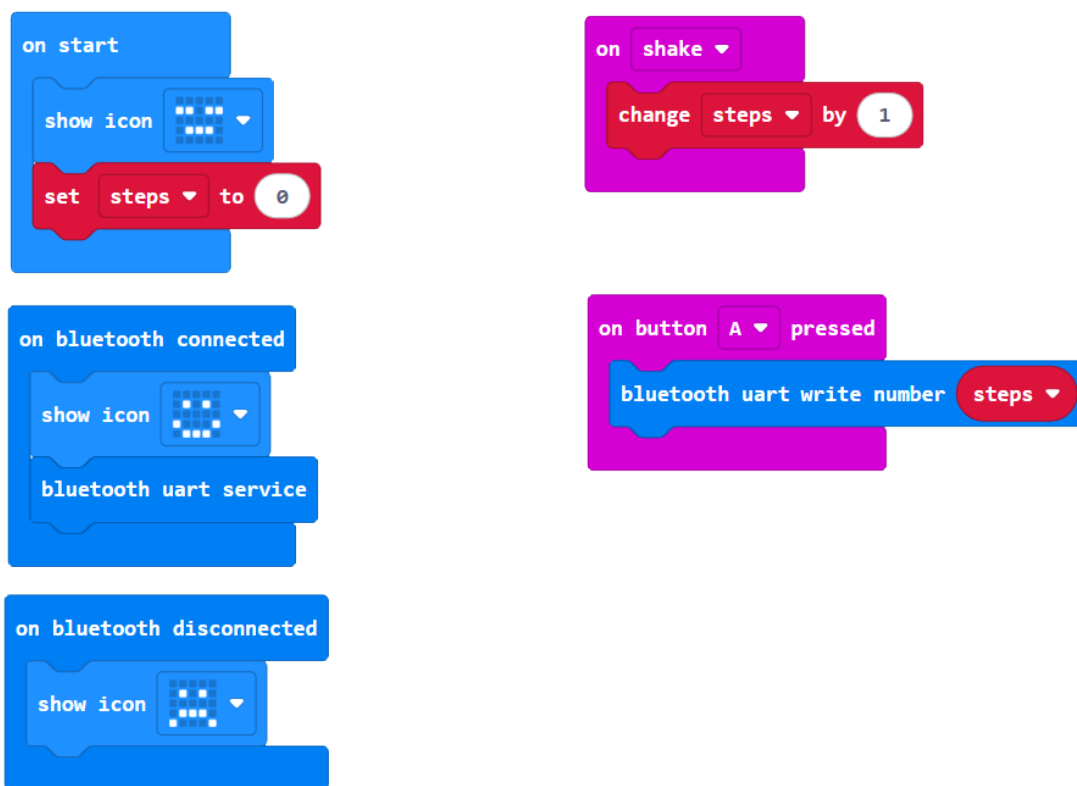
Można także dodać polecenie SHOW ICON, aby wyświetlić coś w rodzaju uśmiechniętej buźki (lub cokolwiek) na ekranie LED, aby wskazać, że micro:bit pomyślnie połączył się z innym urządzeniem Bluetooth.

Z tego samego powodu (tj. aby mieć wizualne wskazanie stanu łączności micro:bit) można dodać smutną ikonę w poleceniu blokowania ON BLUETOOTH DISCONNECTED i śpiącą twarz w poleceniu ON START polecenie blokowe.



Ostatnim krokiem jest poinstruowanie micro:bit, aby wysłał zliczone kroki po naciśnięciu przycisku A.

Rozwiń polecenie blokowe ON BUTTON A PRESSED z grupy poleceń wejściowych, kliknij BLUETOOTH UART WRITE NUMBER i umieść zmienną STEPS w polu wartości. Ostateczny skrypt będzie wyglądał następująco:



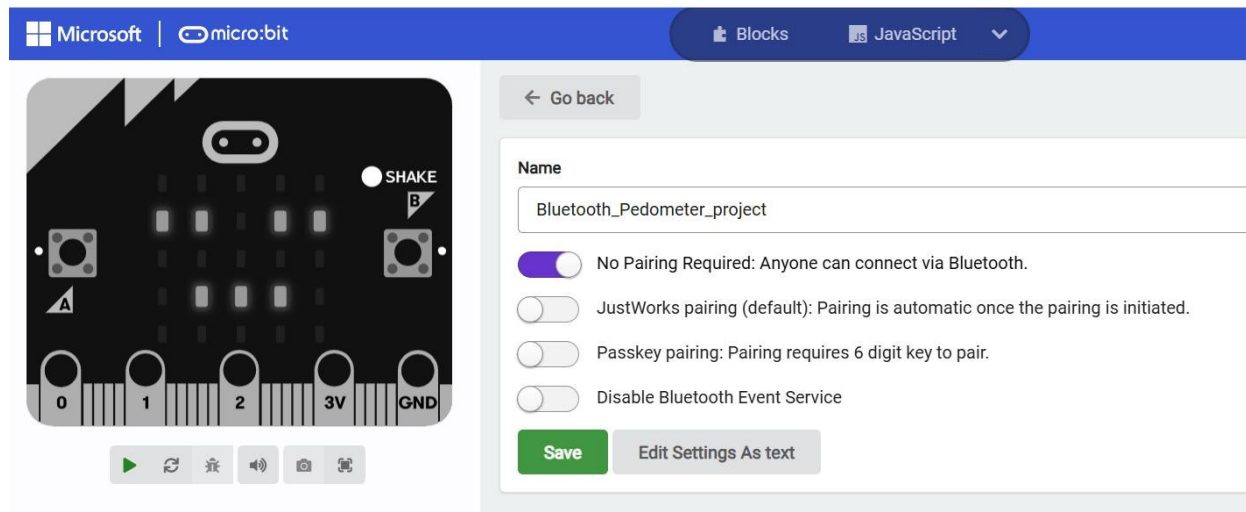
Rysunek 5: Finałowy skrypt do poziomu 1

Skrypt można pobrać do micro:bit.

### Ważne uwagi:

A. Przed przesłaniem skryptu upewnij się, że opcja NO PAIRING REQUIRED: ANYONE CAN CONNECT VIA BLUETOOTH jest wybrana (Zdjęcie 6).

Opcje parowania znajdziesz, klikając menu MORE (tj. ikonę koła zębatego) i wybierając PROJECT SETTINGS z płynającego menu.



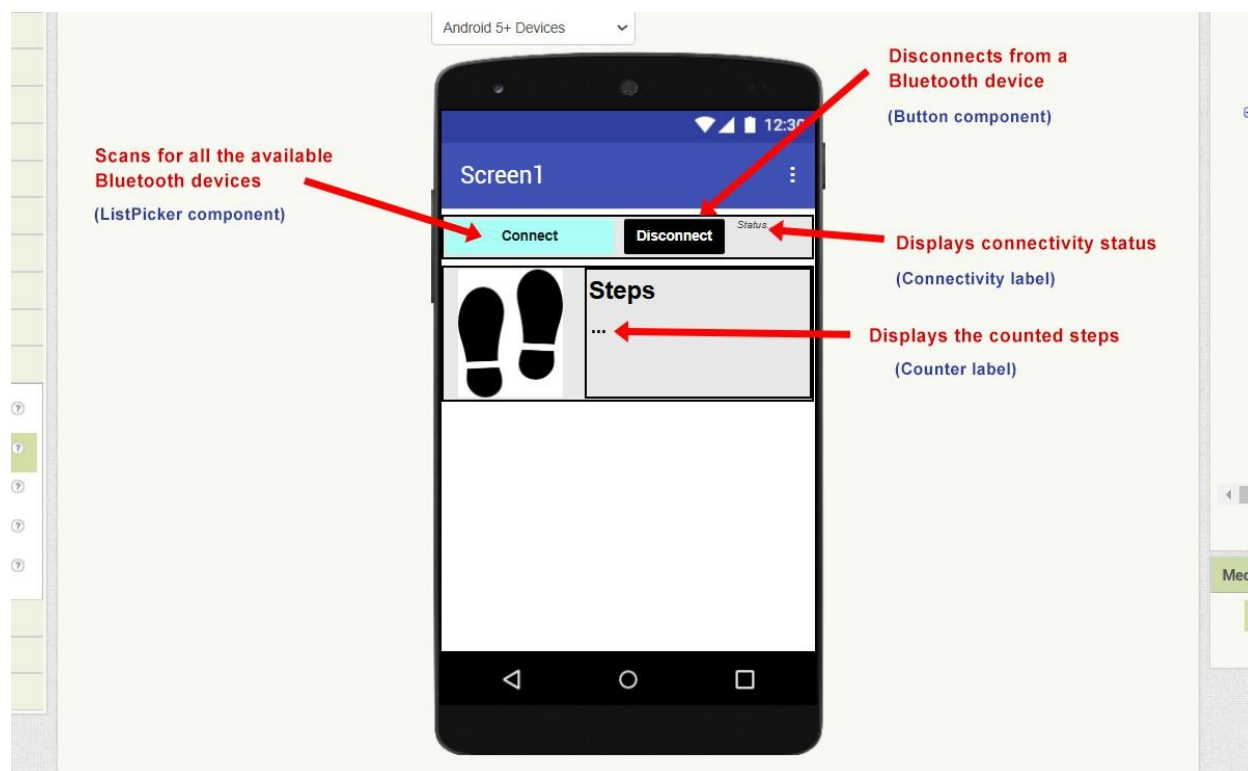
Rysunek 6: Opcje parowania

B. Jeśli micro:bit nie może połączyć się z innym urządzeniem Bluetooth, należy dodać słowo „Bluetooth” do początku nazwy projektu i ponownie przesłać skrypt.

### Część programistyczna MIT App Inventor

Oprócz stworzenia urządzenia z krokomierzem konieczne będzie również opracowanie aplikacji, która będzie go odbierać i wyświetlać zliczone kroki. W tej części należy skorzystać z oprogramowania MIT App Inventor (<https://appinventor.mit.edu/>). Ogólnie rzecz biorąc, tworzenie aplikacji dzieli się na dwie części: projekt interfejsu oraz programowanie dołączonych komponentów. Dokument ten skupi się na części programistycznej, ponieważ część projektowa może być czasochłonna i wydłużyć czas potrzebny na wdrożenie projektu w klasie. Dlatego można pobrać i pracować nad tym plikiem, który zawiera aplikację z już zaprojektowanym interfejsem. Jednakże, w zależności od dostępnego czasu zajęć i poziomu uczniów, można tworzyć aplikację od podstaw. W tym wypadku poniżej znajdują się szczegółowe wytyczne jak zaprojektować interfejs programu aplikacja.

Poniższy obraz (Zdjęcie 7) przedstawia komponenty, które należy zaprogramować.



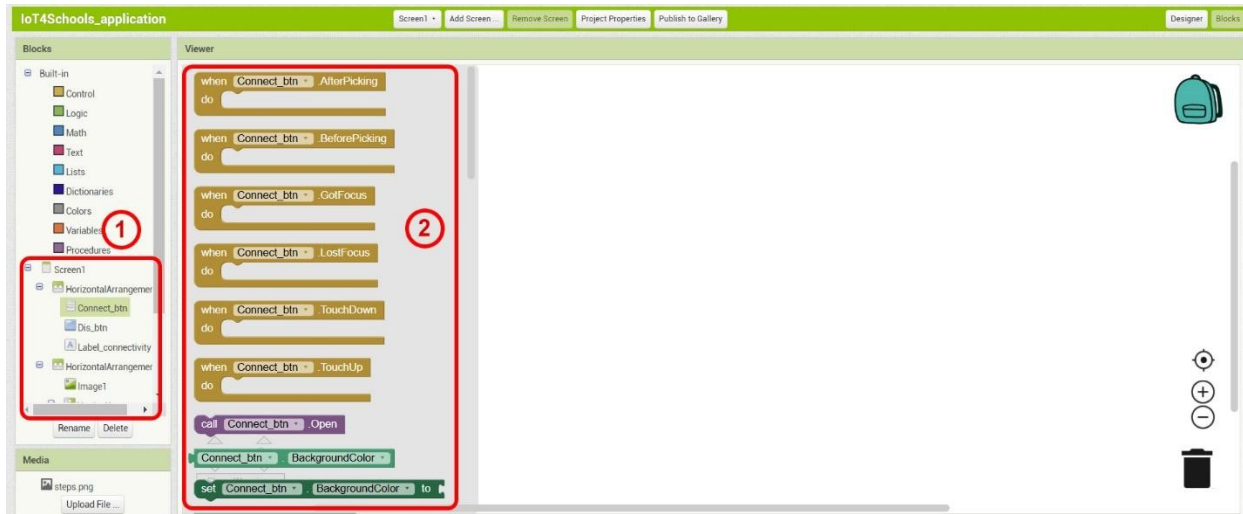
*Rysunek 7: Komponenty, które powinny być oprogramowane*

W szczególności należy zaprogramować:

- przycisk CONNECT (który jest komponentem ListPicker), aby wyszukać wszystkie dostępne moduły Bluetooth LE urządzeń, wyświetlić je w formie listy i pozwolić użytkownikowi wybrać to, którego chce używać (w naszym przypadku urządzenie micro:bit).
- przycisk DISCONNECT (który jest składnikiem przycisku), aby odłączyć aplikację od podłączonego urządzenia Bluetooth
- STATUS (który jest składnikiem etykiety), aby w zależności od tego zmienić się na CONNECTED lub DISCONNECTED w zależności od stanu połączenia.
- „...” (który jest również elementem etykiety), aby wyświetlić zliczone kroki za każdym razem, gdy zostanie naciśnięty przycisk A w micro:bit.

#### **Wskazówka:**

Aby zaprogramować komponent w MIT App Inventor, przejdź do menu BLOCKS i wybierz żądany komponent do programu z listy BLOCKS (1) (Zdjęcie 8). Menu, które się pojawi zawiera wszystkie dostępne polecenia do programowania każdego komponentu (2). Znajdź potrzebne polecenie i przeciągnij je do obszaru przeglądarki, gdzie możesz złożyć swój skrypt.



Rysunek 8: Znalezienie bloku poleceń do zaprogramowania komponentu

## Programowanie przycisków CONNECT i DISCONNECT

**Uwaga:** zapoznanie uczniów z tą częścią programowania nie jest obowiązkowe. W zależności od ich poziomu, możesz zdecydować, czy je wprowadzić, czy bezpośrednio udostępnić im rozwiązanie programistyczne.

### Przycisk CONNECT

Poniższy diagram przedstawia ogólną koncepcję programowania przycisku Connect (tj.komponent ListPicker).



Na tej podstawie musisz użyć:

- polecenie (BeforePicking), które instruuje usługę Bluetooth, aby wyszukała dostępne Bluetooth adresy
- polecenie (Set), które wypełni listę komponentu ListPicker dostępnym modulem Bluetooth adresy
- polecenie (AfterPicking), które zatrzyma skanowanie i połączy aplikację z wybranym adresem Bluetooth.

Dlatego wybierz komponent „Connect\_btn” i z pływającego menu wybierz polecenie zdarzenia

when Connect\_btn BeforePicking, Następnie wybierz komponent BluetoothLE1 i wybierz opcję call BluetoothLE1 StartScanning i umieść go w poprzednim poleceniu.

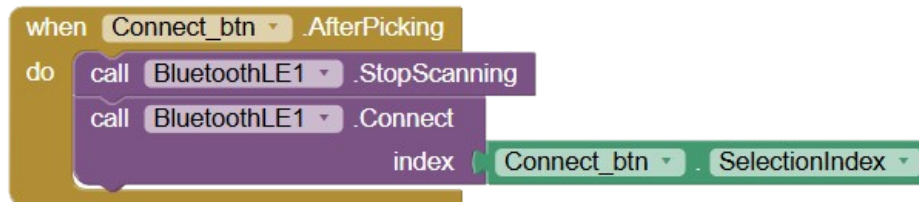


Następnie wybierz komponent BluetoothLE1 i polecenie When BluetoothLE1.DeviceFound.

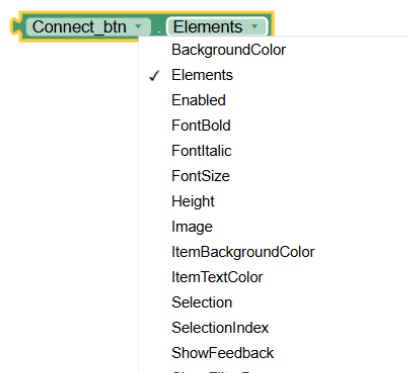
Wewnątrz tego polecenia umieść polecenie set Connect\_btn.ElementsFromString to, znajdujące się w Menu poleceń Connect\_btn. Następnie wybierz ponownie BluetoothLE1, aby znaleźć polecenie BluetoothLE1.DeviceList i umieść je na końcu poprzedniego polecenia.



Wybierz ponownie opcję „Connect\_btn” i użyj polecenia when Connect\_btn.AfterPickin . W tym poleceniu, umieść polecenia call BluetoothLE1.StopScanning i call BluetoothLE1.Connect indeks, oba znajdują się w menu poleceń BluetoothLE1. Następnie z zakładki index wybierz polecenie Connect\_btn.SelectionIndex .



**Uwaga:** SelectionIndex jest jedną z dostępnych opcji polecenia Connect\_Btn Elements.

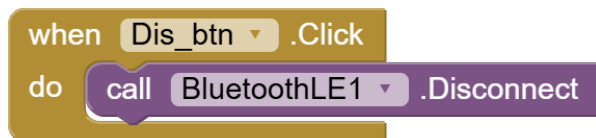


Skrypt umożliwiający połączenie Twojej aplikacji z micro:bit jest już gotowy.

### Przycisk DISCONNECT

Naciśnięcie przycisku DISCONNECT zakończy połączenie z wybranym urządzeniem Bluetooth.

Dlatego wybierz komponent Dis\_btn i przeciągnij polecenie When Dis\_btn.Click , a następnie wybierz polecenie call BluetoothLE1.Disconnect, znajdujące się w menu BluetoothLE1.

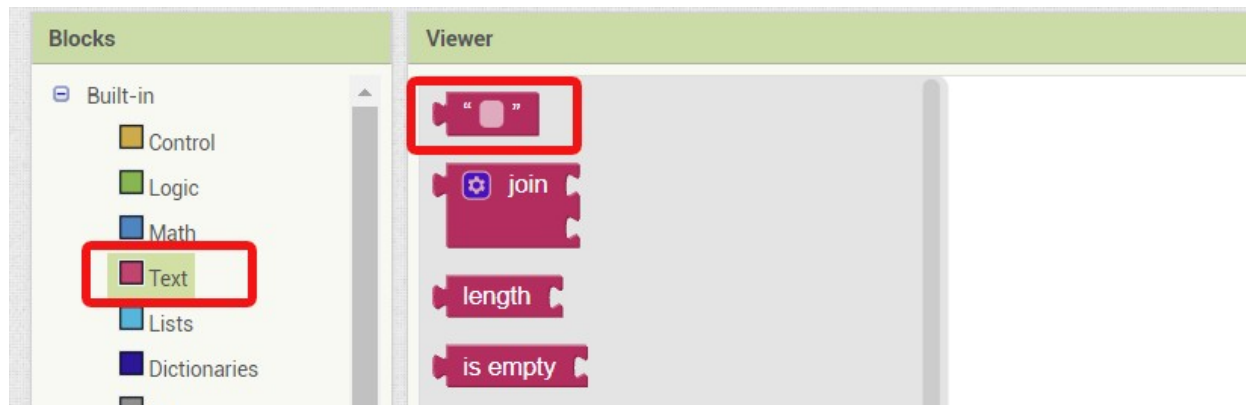


Skrypt przycisku DISCONNECT jest już gotowy.

Oprócz dwóch przycisków znajduje się tam również etykieta wskazująca, czy połączenie zostało nawiązane.

Aby to zrobić, należy utworzyć następujące skrypty:

Wybierz komponent „Bluetooth” i przeciągnij polecenie „When Bluetooth.Connected do”. Następnie wybierz Komponent Label\_connectivity i użyć polecenia set Label\_connectivity.Text to. Następnie przejść do menu poleceń tekstowych, przeciągnąć puste polecenie wprowadzania tekstu (Zdjęcie 9) i przyciągnij je obok pliku set Label\_connectivity.Text to. Następnie wpisz słowo Connected .



Rysunek 9: Znajdywanie polecenia do wprowadzania tekstu

Teraz, gdy aplikacja jest podłączona do micro:bit, słowo STATUS zmieni się na CONNECTED .



Aby zmienić słowo STATUS na DISCONNECTED po zakończeniu połączenia należy dodać podobny wiersz kodu do polecenia When Dis\_btn.Clic”.



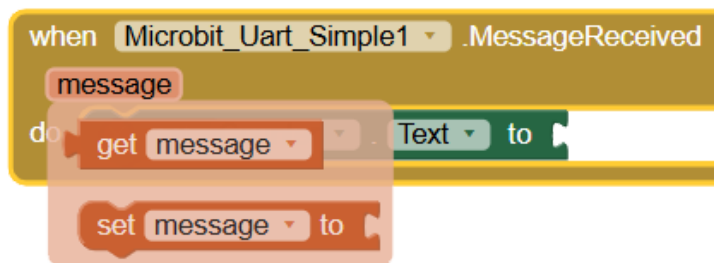
set Label\_connectivity . Text to "Disconnected"

W ten sposób naciśnięcie przycisku DISCONNECT spowoduje zakończenie połączenia Bluetooth a słowo STATUS zmieni się na DISCONNECTED.

### Zaprogramowanie aplikacji do odbierania komunikatów z urządzenia krokomierza

Głównym celem, z programistycznego punktu widzenia, jest to, aby uczniowie nauczyli się, jak projektować aplikację, która może komunikować się i wymieniać dane z micro:bit (tj. urządzenie z krokomierzem). Programując micro:bit, zaprogramowałeś przycisk A tak, aby po jego naciśnięciu wysłał wartość numeryczną (w tym projekcie, liczbę zliczonych kroków) przez Bluetooth. Po podłączeniu do krokomierza, nasza aplikacja może odbierać i wyświetlać tę wartość liczbową. Etykieta wyświetlająca tę wartość to „Licznik - COUNTER”. Należy zatem zaprogramować aplikację tak, aby zmieniała treść tej etykiety na liczbę otrzymanych kroków. Można tego dokonać poprzez zaprogramowanie komponentu Microbit\_Uart\_Simple1.

Wybierz komponent Microbit\_Uart\_Simple1 i przeciągnij plik when Microbit\_Uart\_Simple1.MessageReceived do do przeglądarki. Następnie wybierz COUNTER i przeciągnij polecenie set Counter.Text to do powyższego polecenia. Najedź kursorem na pole wiadomości, a następnie z pływającego menu (zdjęcie 10) przeciągnij polecenie get message i wybierz polecenie set Counter.Text to .



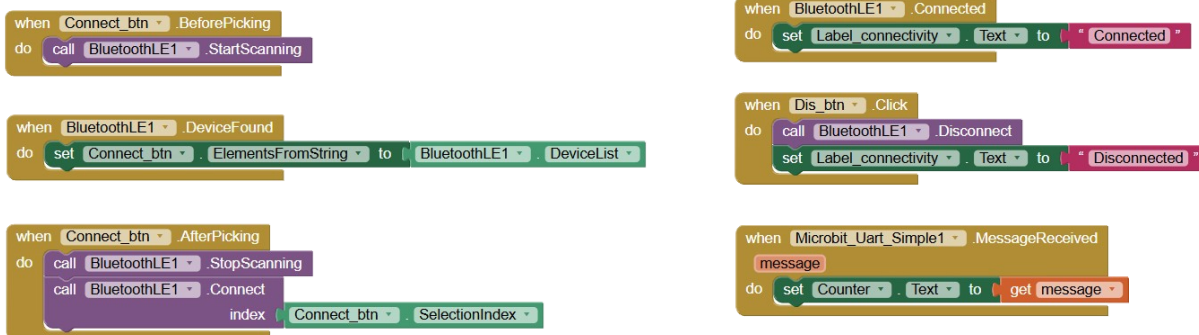
Rysunek 10: Znajdywanie komendy "get message"

Dzięki temu skryptowi aplikacja wyświetli każdorazowo liczbę kroków zliczoną przez użytkownika krokomierza , po naciśnięciu przycisku A.



Poniższy obraz przedstawia cały skrypt aplikacji.



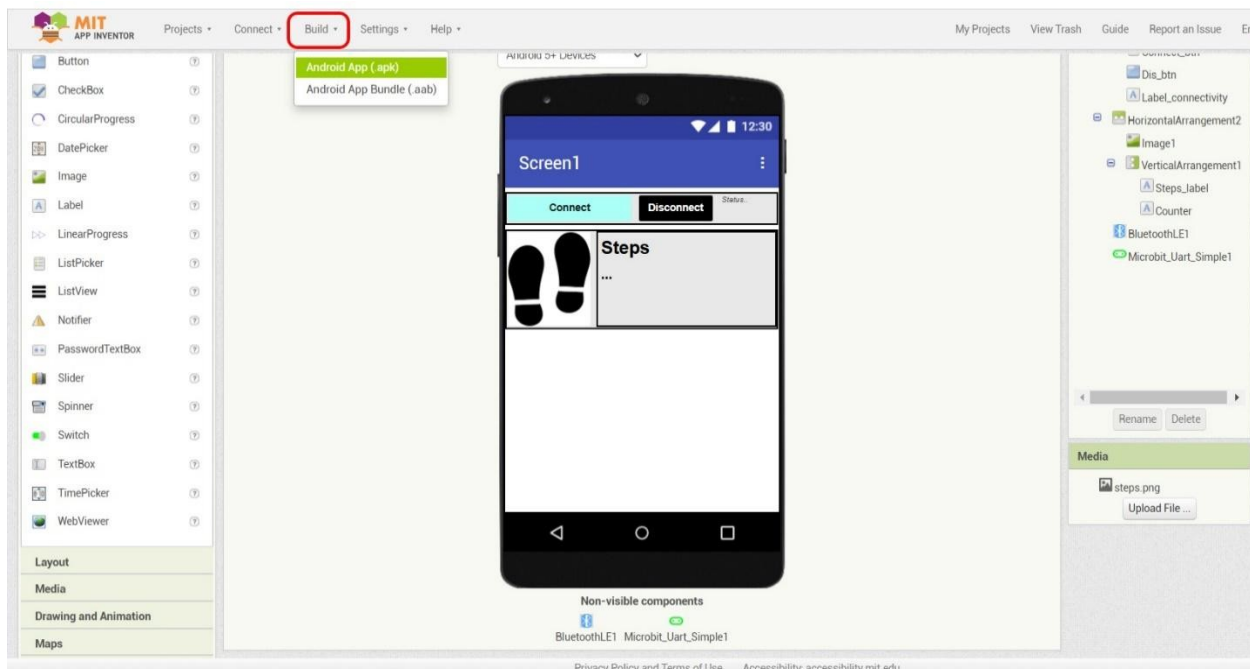


Aplikacja jest teraz zaprogramowana i gotowa do zainstalowania na urządzeniu inteligentnym, takim jak smartfon.

### Budowanie aplikacji

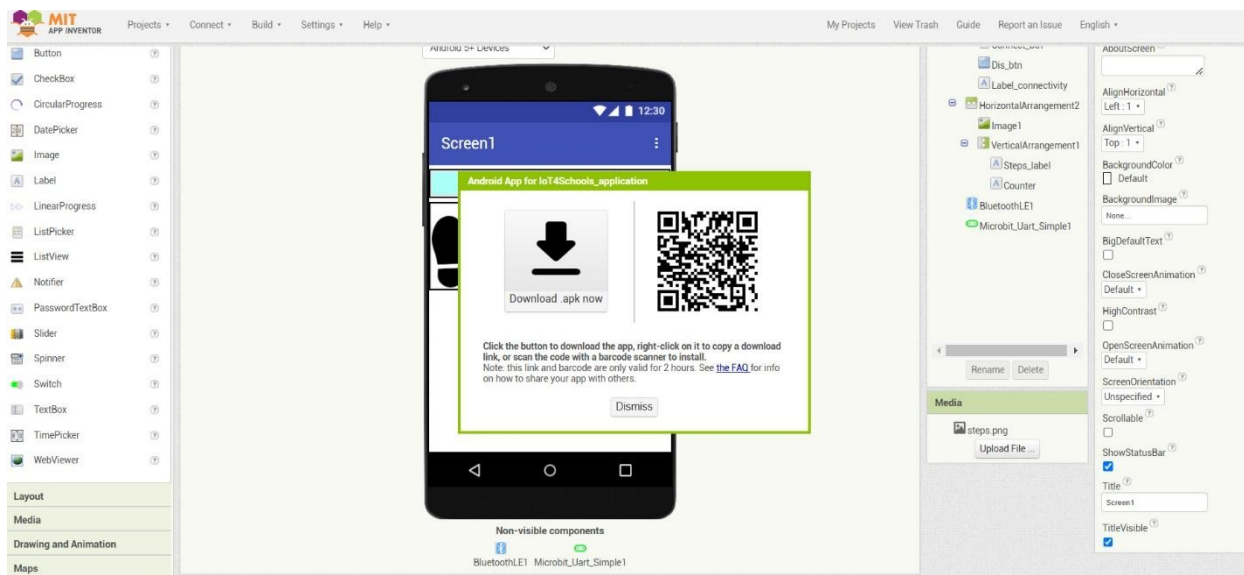
Aby zainstalować aplikację na urządzeniu inteligentnym, należy ją najpierw zbudować.

Kliknij z górnego menu ikonę Build (Zdjęcie 11) i wybierz opcję Aplikacja na Androida (apk) z pływającego menu.



Rysunek 11: Tworzenie aplikacji

Może to potrwać kilka minut. Po zakończeniu procesu możesz pobrać plik .apk lub zeskanować go za pomocą urządzenia inteligentnego, korzystając z aplikacji MIT AI2 Companion (Zdjęcie 12).



Rysunek 12: Pobierz plik .apk lub zeskanuj, aby zainstalować aplikację

**Uwaga 1:** MIT AI2 Companion to aplikacja jest dostępna bezpłatnie i można ją znaleźć w sklepie „Play” na Twoim urządzeniu smart. Ta aplikacja jest medium, które zapewnia powodzenie w instalacji wygenerowanego pliku

**Uwaga 2:** Podczas instalacji pliku .apk możesz otrzymać wiele komunikatów dotyczących bezpieczeństwa tego pliku. Zignoruj je wszystkie i poproś urządzenie, aby kontynuowało proces instalacji.

### Parowanie aplikacji z krokomierzem

Jak wspomniano, aby połączyć aplikację z krokomierzem, należy nacisnąć przycisk CONNECT i wybierz z listy adres Bluetooth micro:bit. Jednak prawdopodobne jest, że micro:bit nie będzie umieszczone na liście dostępnych urządzeń Bluetooth. Aby rozwiązać ten problem, wróć do głównego menu programu i włącz na kilka sekund tryb „samolotowy” na swoim urządzeniu mobilnym. Następnie dezaktywuj go i ponownie naciśnij przycisk Skanuj. Adres Bluetooth Micro:bit będzie teraz dostępny.

**Uwaga:** upewnij się, że opcja „Lokalizacja” jest również włączona na Twoim urządzeniu smart .

### 2.1.3 Crafting

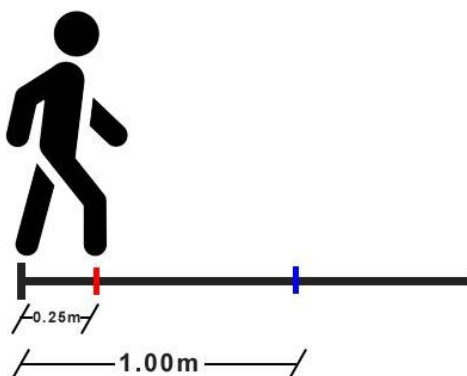
Zachęć uczniów, aby znaleźli sposoby na podłączenie micro:bit do zewnętrznego źródła zasilania. Zaproponuj im, aby przymocowali krokomierz do różnych części ciała (ręce, nogi itp.) i obserwuj, czy ma to wpływ na wyniki.

## 2.2 Poziom 2: Programowanie krokomierza do rejestrowania wielu danych

### 2.2.1 Mierzenie przebytej odległości z wykorzystaniem krokomierza

Na tym poziomie uczniowie dowiedzą się, jak wykonać bardziej zaawansowany krokomierz, programując urządzenie aby mógł zmierzyć dystans przebytej podczas chodzenia. W tym celu będą mierzyć odległość na podstawie liczby wykonanych kroków.

Najpierw zachęć ich do zmierzenia odległości, robiąc tylko jeden krok. Załóżmy, że dana osoba przebyła 25 cm, wykonując tylko jeden krok (jak pokazano na poniższym schemacie). Na podstawie tego pomiaru ta osoba musiałaby wykonać około 4 kroków, aby pokonać odległość 1 metra.



Uznając to za podstawę, uczniowie wprowadzą następujące zmiany w skrypcie micro:bit:

### 2.2.2 Programowanie

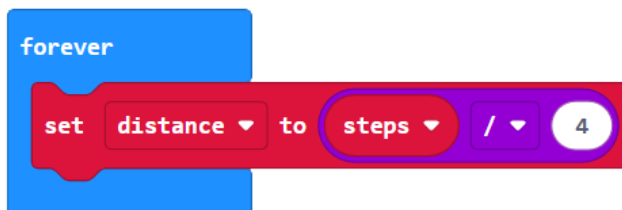
#### Część programistyczna Makecode

Kontynuuj pracę nad poprzednim skryptem, ale najpierw zapisz bieżącą pracę jako nowy plik. Najpierw utwórz nową zmienną dla odległości. Następnie dodaj polecenie set distance to 0 do skryptu on start, aby zainicjować odległość.



Następnie dodaj pętlę forever, w której umieścisz skrypt obliczający odległość na podstawie zliczonych kroków, jak wspomniano w poprzednim przykładzie (tj. wykres przebytej odległości).

Dodaj polecenie set distance to... i w polu wartości dodaj polecenie dzielenia (.../...) z narzędzia Math. Następnie w polu dzielnej dodaj zmienne kroki, a w polu dzielnika przybliżoną wartość liczby kroków potrzebnych do pokonania dystansu 1 metra (np. cyfra 4, jeżeli dystans przebyty w jednym kroku wynosi 0,25m) .

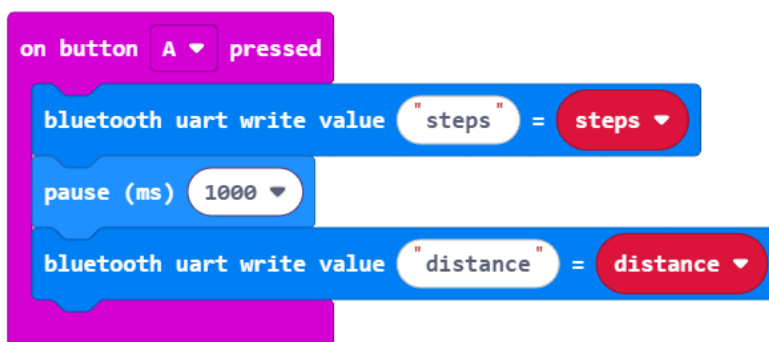


Ostatnim krokiem jest przesłanie tych danych do aplikacji po naciśnięciu przycisku A.

Obecnie micro:bit nie może wysłać wielu danych jednocześnie do aplikacji opracowanej przy użyciu aplikacji MIT Inventor. Możliwe jest wysłanie wielu danych, wysyłając jedno dane po drugim.

W poleceniu on button A pressed zastąp polecenie „Bluetooth uart write number” poleceniem „Bluetooth uart write value „x” = 0”. W polach „x” wpisz „kroki” i „odległość”.

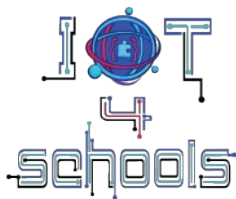
Pomiędzy tymi poleceniami dodaj polecenie „pauza” i ustaw czas trwania na 1000 ms (tj. 1 sekundę).



Pobierz zmodyfikowany skrypt na micro:bit i przetestuj wyniki.

### Część programistyczna MIT App Inventor

Nie musisz wprowadzać żadnych zmian w skrypcie w App Inventor. Dla spójności możesz po prostu zmienić tekst komponentu Steps\_label (w menu Designer) na „Kroki/Odległość” lub „pomiar” lub „dane dotyczące kondycji”.



### 3 Wskazówki i zalecenia

#### 3.1 Dalszy rozwój projektu

W zależności od poziomu uczniów i dostępnego czasu, możesz rozszerzyć projekt o pozyskiwanie dodatkowych danych dotyczących kondycji i zdrowia, dodając inne zmienne, takie jak kalorie spalone podczas chodzenia (szacuje się, że człowiek spala około 0,04 do 0,05 kalorii na krok) lub poprzez dodanie zewnętrznych czujników taki jak czujnik tętna (na przykład czujnik tętna Gravity Heart Rate Monitor, który jest kompatybilny z mikro:bit).

#### 3.2 Personalizacja krokomierza

Można również zachęcić uczniów do tworzenia własnych, spersonalizowanych krokomierzy (w oparciu o dystans, który pokonują jednym krokiem), a potem udostępniają swoje urządzenia innym rówieśnikom, aby sprawdzić, czy są jakieś różnice w otrzymanych danych.

#### 3.3 Sprawdzenie czy micro:bit można połączyć poprzez Bluetooth

Zanim spróbujesz połączyć micro:bit z utworzoną aplikacją, upewnij się, że twoje urządzenie smart faktycznie rozpoznaje micro:bit. Aby to zrobić, otwórz menu Bluetooth na swoim urządzeniu smart i sprawdź, czy na liście dostępnych połączeń pojawi się micro:bit.

#### 3.4 Omówienie zalet i wad

Po zakończeniu projektu zachęć uczniów, aby podzielili się swoimi przemyśleniami i pomysłami na temat zalet i wad korzystania z urządzeń takich jak krokomierz w codziennym życiu, nawyków związanych z aktywnością fizyczną i problemami zdrowotnymi. Można ich również zapytać co sądzą o udostępnianiu takich danych w chmurze oraz czy uważają, że istnieje jakiekolwiek ryzyko w zakresie danych osobowych oraz bezpieczeństwa.

### 4 Materiał źródłowy

[1] Pedometer: <https://en.wikipedia.org/wiki/Pedometer>

[2] Microsoft Makecode software: <https://makecode.microbit.org/>

[3] MIT App Inventor: <https://appinventor.mit.edu/>

[4] MIT AI2 Companion App: <https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3&hl=en>

[5] Gravity Heart Rate Monitor Sensor: [https://grobotronics.com/gravity-heart-rate-monitor-sensor-ppg.html?sl=en&srsId=AfmBOoo77f0hnby0h2SwewfcRUGdHkYM5Xfm\\_NB7vAAWGHf5b2kaWUKB](https://grobotronics.com/gravity-heart-rate-monitor-sensor-ppg.html?sl=en&srsId=AfmBOoo77f0hnby0h2SwewfcRUGdHkYM5Xfm_NB7vAAWGHf5b2kaWUKB)