



Lekcja 14. Projekty własne i App Inventor

Cel lekcji

Celem lekcji jest wprowadzenie w platformę App Inventor i zaprezentowanie możliwości środowiska oraz rozwijanie pracy kreatywnej. Nauczyciel wykonuje projekt angażując uczestników poprzez pytania, zachęcanie do aktywności i wpływ na wygląd projektu. Na tę część przeznaczamy nie więcej niż 45 min, tworząc projekt na oczach uczestników. Podczas pierwszej części zajęć uczestnicy wykonują projekty własne w Mine-imatorze.

Agenda

1. Pytania wstępne i rozmowa
2. Praca kreatywna w Minecrafcie
3. Stworzenie aplikacji w App Inventorze
4. Podsumowanie lekcji i pytania końcowe

Część I

Rozpoczynamy od pracy kreatywnej w Mineimatorze. Uczniowie mają 45 min na to, aby stworzyć prostą animację. Tematyka animacji jest dowolna, jednak konieczne jest, aby pojawiły się w niej elementy z poniższej listy. Po wykonaniu projektów uczniowie pokazują swoje animacje innym uczestnikom i opowiadają co udało im się zrobić.

Elementy konieczne w projekcie:

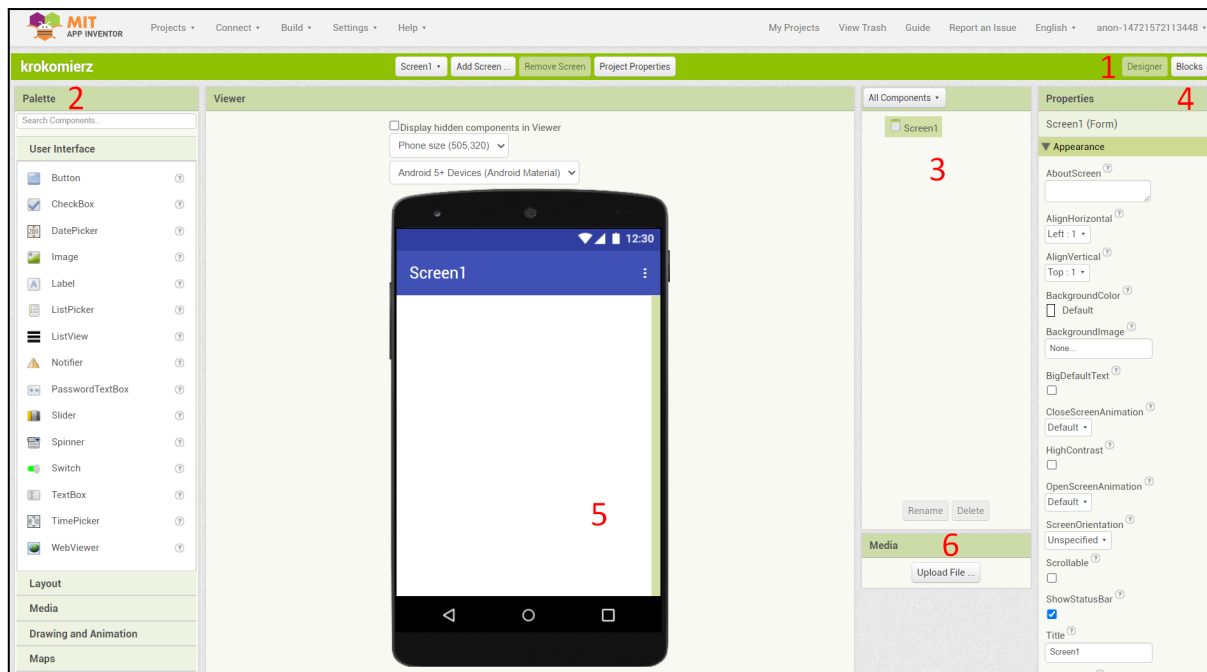
- minimum 1 mob
- minimum 2 animowane części ciała moba
- zmiana koloru podłoża
- dodanie źródła światła
- dodanie kamery
- dodanie 10 obiektów ozdobnych (niekoniecznie animowanych)
- dodanie jednego dowolnego kształtu

Podczas pracy kreatywnej uczestników należy aktywizować dzieci i angażować je do pracy. Na bieżąco sprawdzamy jak sobie radzą, zadajemy pytania, podsuwamy pomysły. Dzieci muszą czuć zaangażowanie ze strony trenera. Nie szczędzimy też pochwał, zarówno w trakcie pracy, jak i podczas prezentowania dzieł uczniów.

Część II - wykonuje tylko trener

Pokazanie możliwości i funkcji platformy App Inventor

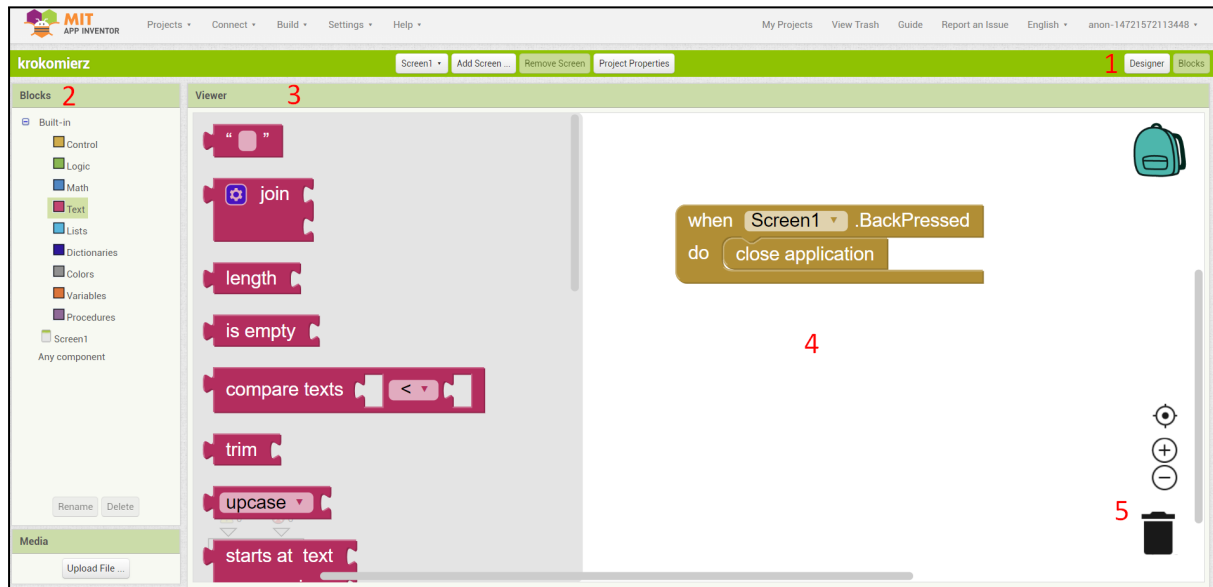
1. Wchodzimy na stronę: <http://code.appinventor.mit.edu/>
(kod który się pojawi zapisujemy - dzięki niemu możemy wrócić do projektu)
2. Klikamy przycisk "Start new project", i nadajemy mu nazwę
3. Omówienie elementów na ekranie (Designer - strona startowa)



1. **Designer/Blocks** (*Projektowanie/Bloki*) - praca nad aplikacją odbywa się w jednym z dwóch obszarów. Pierwszy z nich, Designer, to miejsce, gdzie projektujemy ogólny wygląd i interfejs aplikacji. Drugi, Blocks, to z kolei obszar, w którym pracujemy nad budową i działaniem aplikacji. O tym, w jakim trybie aktualnie się znajdujemy, informuje nas niebieskie podświetlenie danego przycisku. W App Inventorze, inaczej niż w Scratchu, praca nad wyglądem oraz nad działaniem aplikacji jest rozdzielona. Co więcej, narzędzia przeznaczone do pracy nad wyglądem aplikacji są znacznie bardziej rozbudowane, niż w przypadku wspomnianego już Scratcha.
2. **Palette** (*Paleta*) - zbiór narzędzi przeznaczonych do budowania aplikacji i pracy nad jej warstwą wizualną. Wszystkie narzędzia zostały pogrupowane w odpowiednie kategorie (np. User Interface, Layout, Media). Ich użycie odbywa się na zasadzie “drag & drop”, czyli “przeciągnij i upuść”.
3. **Components** (*Komponenty*) - w tym miejscu zarządzamy wszystkimi komponentami, z jakich składa się nasza aplikacja. Możemy tutaj zmieniać ich nazwy (Rename) bądź też usuwać (Delete). Zaznaczenie odpowiedniego komponentu pozwala również na jego konfigurację w oknie Properties.</>
4. **Properties** (*Właściwości*) - ten element pozwala na ustawienie właściwości wybranego komponentu. Dostępne w tym oknie funkcje zmieniają się w zależności od wybranego komponentu. Np. dla elementu tekstowego możemy zmienić m.in. krój czcionki, kolor i wielkość oraz parametry widoczności.
5. **Viewer** (*Podgląd*)- w tym miejscu po prostu swobodnie układamy poszczególne komponenty na ekranie.

6. **Media** (*elementy multimedialne*) - tutaj znajdziemy wszystkie pliki multimedialne, które są używane w aplikacji. Będą to zarówno pliki graficzne, jak i dźwiękowe.

5. Omówienie elementów Block



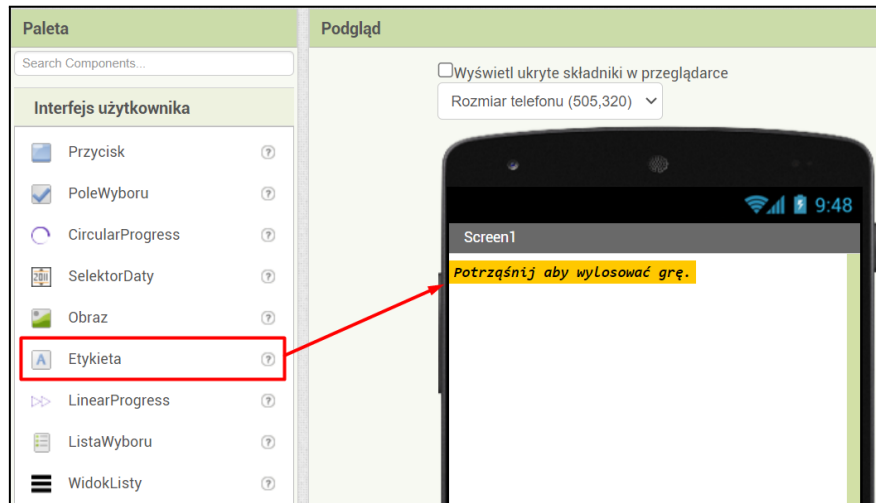
1. **Designer**
2. **Blocks** - podobnie jak w Scratchu, znajdziemy tutaj posortowane w odpowiednie kategorie bloki zawierające różne funkcje. Użycie bloków również odbywa się na zasadzie "drag & drop". Do dyspozycji mamy dwie główne grupy: **Built-in**, która zawiera podstawowe, wbudowane polecenia oraz **Screen1**, gdzie znajdziemy komponenty, które wstawiliśmy w obszarze **Designer**. Po wybraniu danego komponentu będziemy mieli dostęp do poleceń właściwych dla wybranego elementu - np. w przypadku przycisku dysponujemy poleceniem **Click**.
3. Tutaj znajdziemy klocki przyporządkowane do kategorii, którą wybraliśmy w menu **Blocks** (2). Na ilustracji widzimy klocki z kategorii **Text**, która znajduje się w grupie **Built-in**.
4. **Viewer** - w tym oknie budujemy program do naszej aplikacji. Ten element działa tak samo, jak jego odpowiednik w programie Scratch.
5. **Śmietnik** - przeciągając do kosza niepotrzebne bloki z obszaru **Viewer**, usuniemy je.

Przygotowanie aplikacji Maszyna Losująca

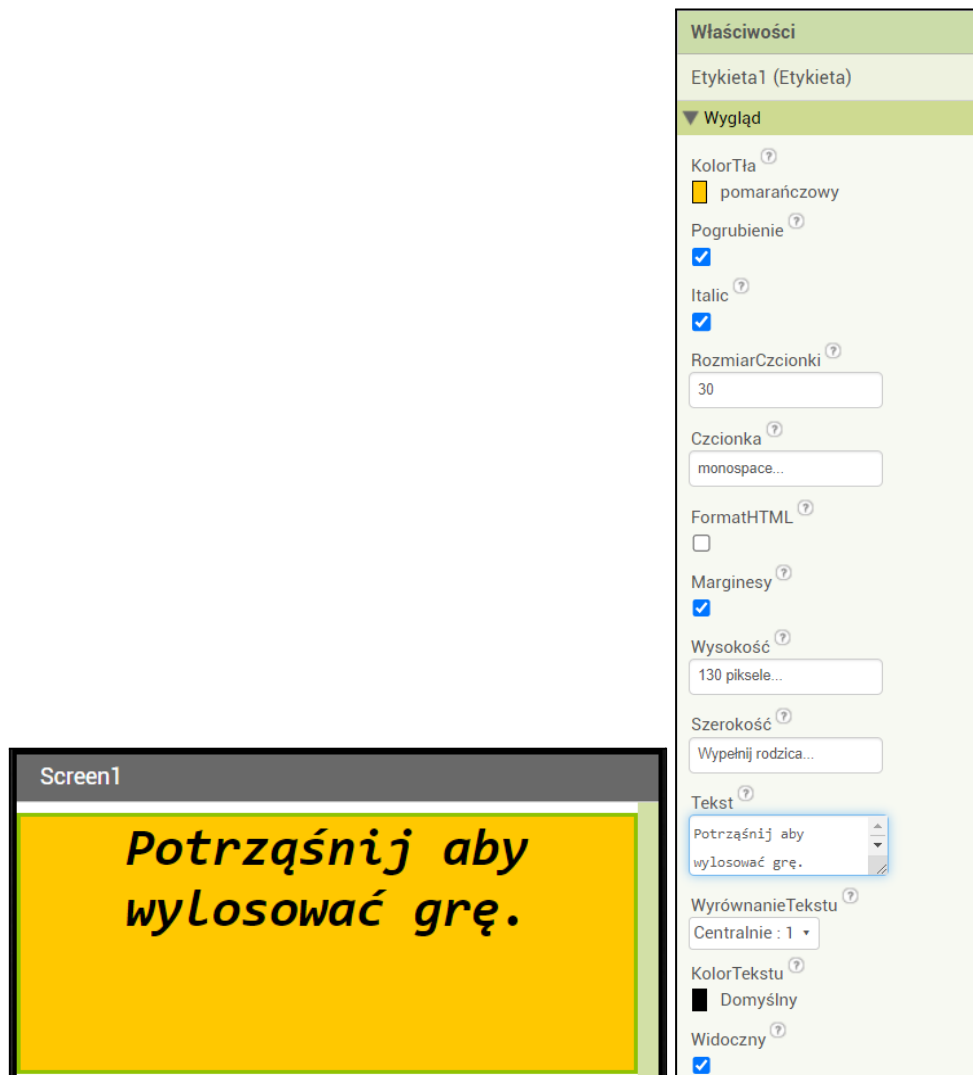
Wgrywamy starter (**Projects > Import project (.aia) from my computer ...**):

https://drive.google.com/file/d/13bIAGRstc1j1papc7fNk5_qfQ97wf5Lm

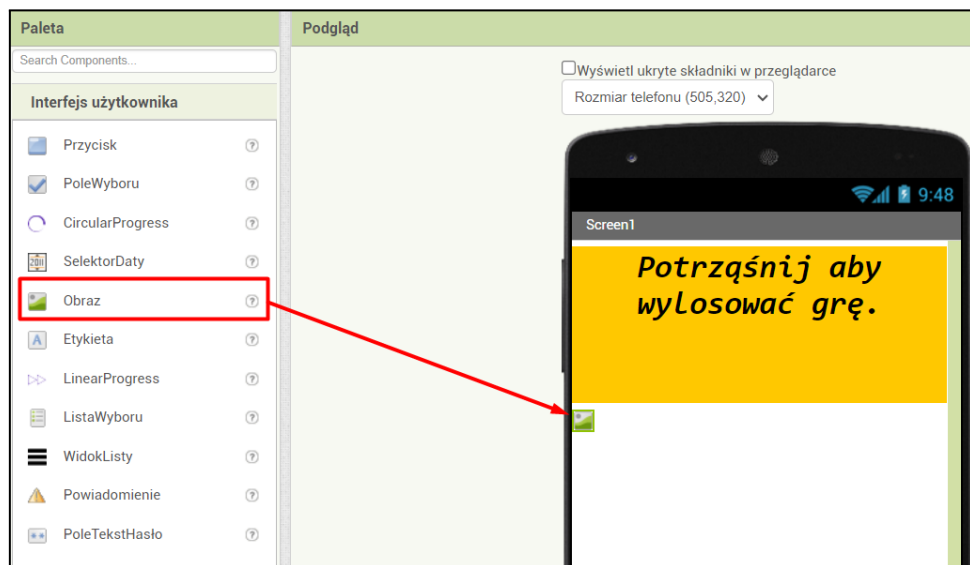
1. Dodajemy na ekran naszej aplikacji etykietę z napisem “Potrząśnij, aby wylosować grę”.



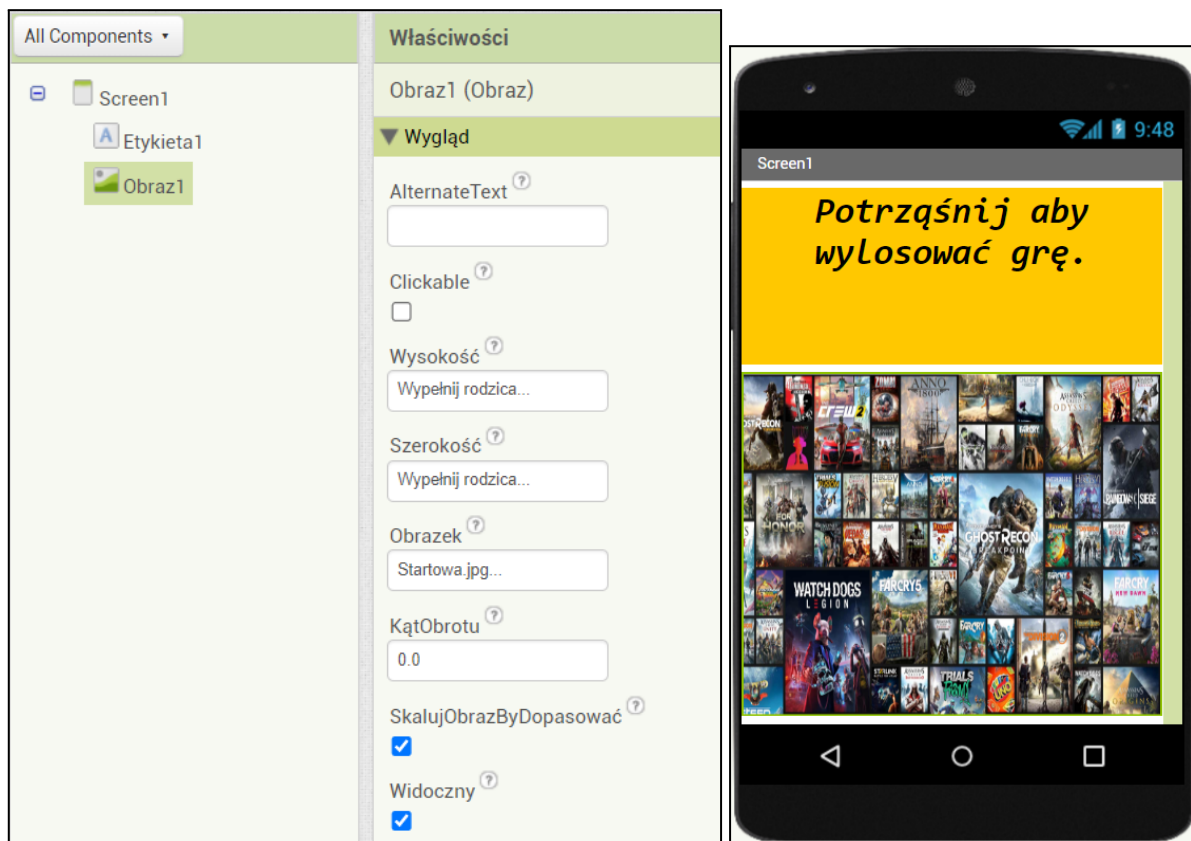
2. Przy pomocy właściwości etykiety edytujemy jej wygląd. Można dowolnie zmienić jej wygląd - poniżej przykładowy sposób. Aby edytować każdy komponent, trzeba go zaznaczyć w sekcji “Składniki”.



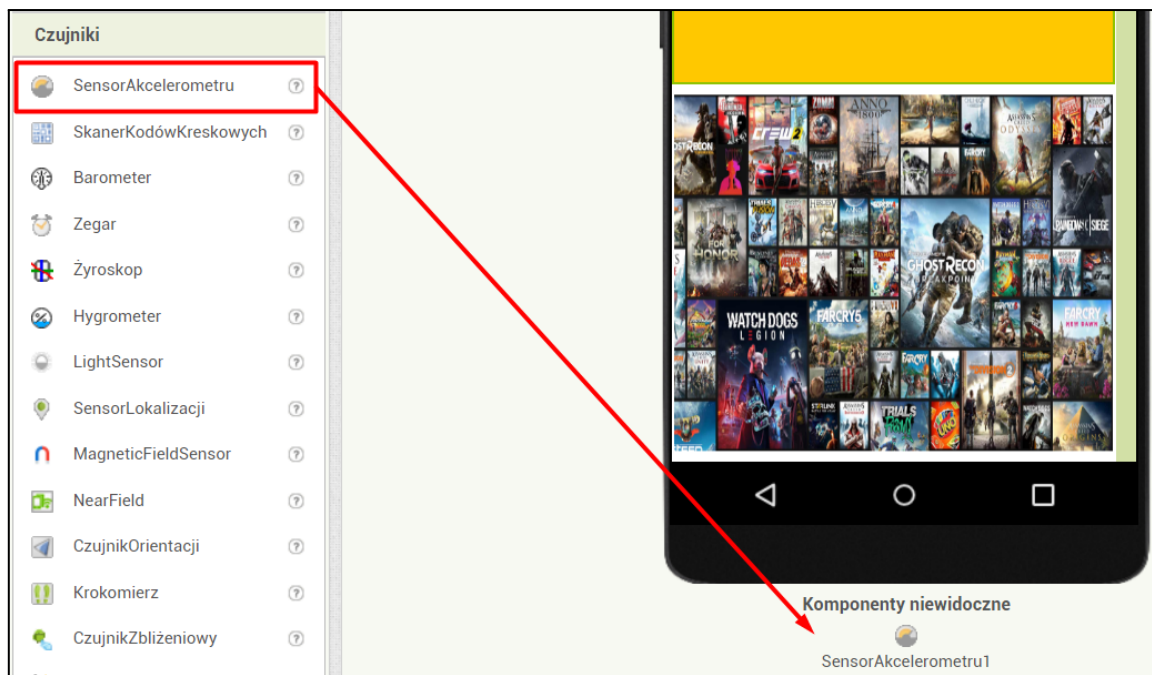
3. Dodajemy pod etykietę Obraz.



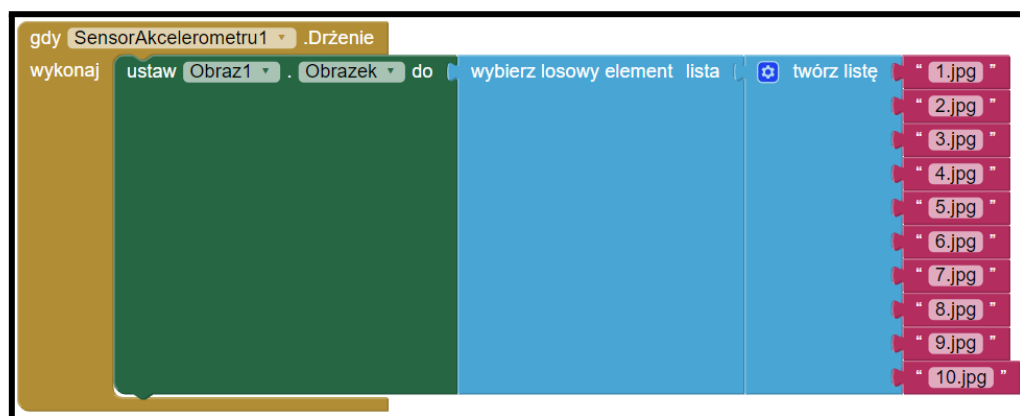
4. W ustawieniach obrazu wybieramy wysokość i szerokość "Wypełnij rodzica". Dodatkowo, w sekcji "Obrazek" wybieramy z załączonych mediów grafikę "Startowa".



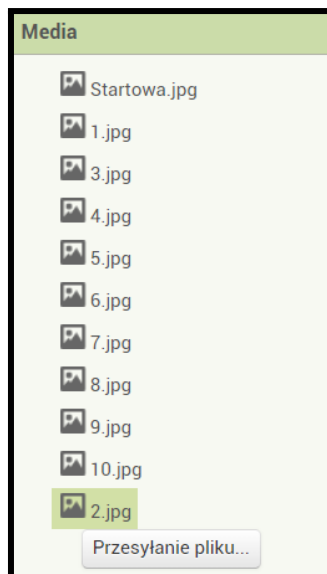
5. Do komponentów dodajemy Sensor Akcelerometru (trzeba upuścić go na telefonie) z kategorii Czujniki. Będzie widoczny pod wizualizacją urządzenia.



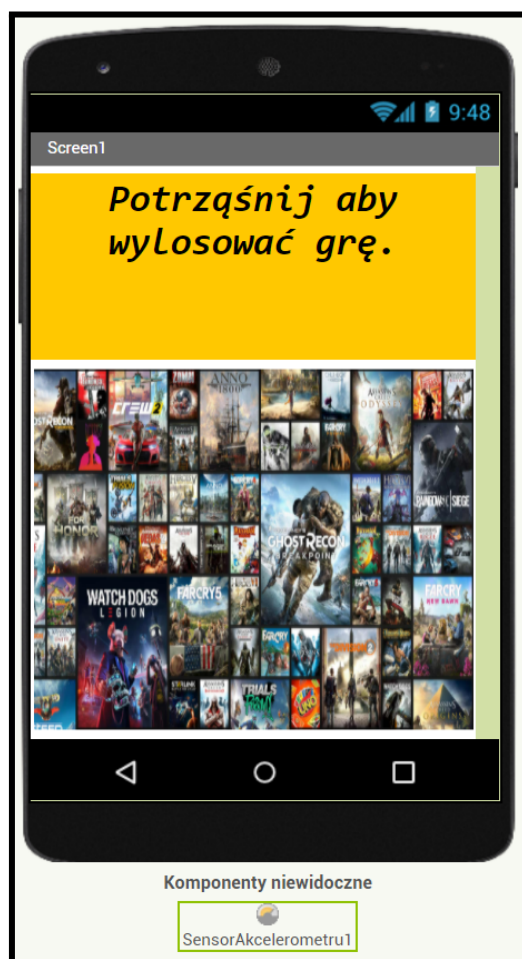
6. Programujemy aplikację do losowania grafiki gry poprzez potrząśnięcie telefonem. Aby dodać więcej elementów do listy, wystarczy kliknąć koło zębate i przeciągnąć bloczek "pozycja" na prawą stronę.



Grafiki znajdują się w mediach.



7. Ostateczny wygląd aplikacji zamieszczono poniżej. Aby wgrać ją na telefon, wystarczy kliknąć "Kompiluj" i "Android App (.apk)". Pojawi się QR Code, dzięki któremu można pobrać apkę.



Testując i prezentując uczestnikom gotową aplikację najlepiej skorzystać z dedykowanej aplikacji MIT AI2 Companion, która ułatwi instalację i testowanie. Dla telefonów z system iOS jest ona obowiązkowa.



MIT AI2 Companion

MIT App Inventor Edukacja

 Dla wszystkich

 Ta aplikacja jest dostępna dla Twojego urządzenia