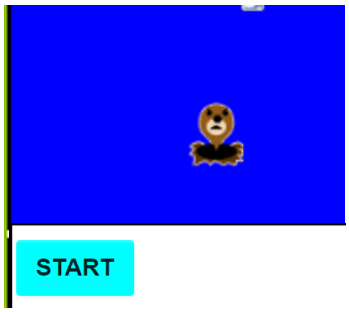
(3) Maulwurf-Spiel (Mole Mash)



Jetzt programmieren wir das Maulwurf-Spiel. Das Ziel ist den Maulwurf zu erwischen, während dieser hin und her hüpft.

Wir zählen, wie oft das klappt (Trefferpunkte) innerhalb von 10 Sekunden Spielzeit.

Wir werden einige neue Komponenten und Blöcke brauchen:

TIPPS:

COMPONENTS

Canvas	*Leinwand. Hier benutzt als Spielfläche.*
Clock:	*Zeitschaltuhr oder Timer. Zählt ein Intervall und gibt am Ende ein Tickerzeichen aus. Kann immer wieder neu mit dem Zählen starten.*
Sprite:	*Eine Spielfigur, welche in deinem Spiel ständig bewegt.*
Notifier:	*Zeigt die BenutzerIn eine Meldung am Ende des Spiels an.*

BLOCKS

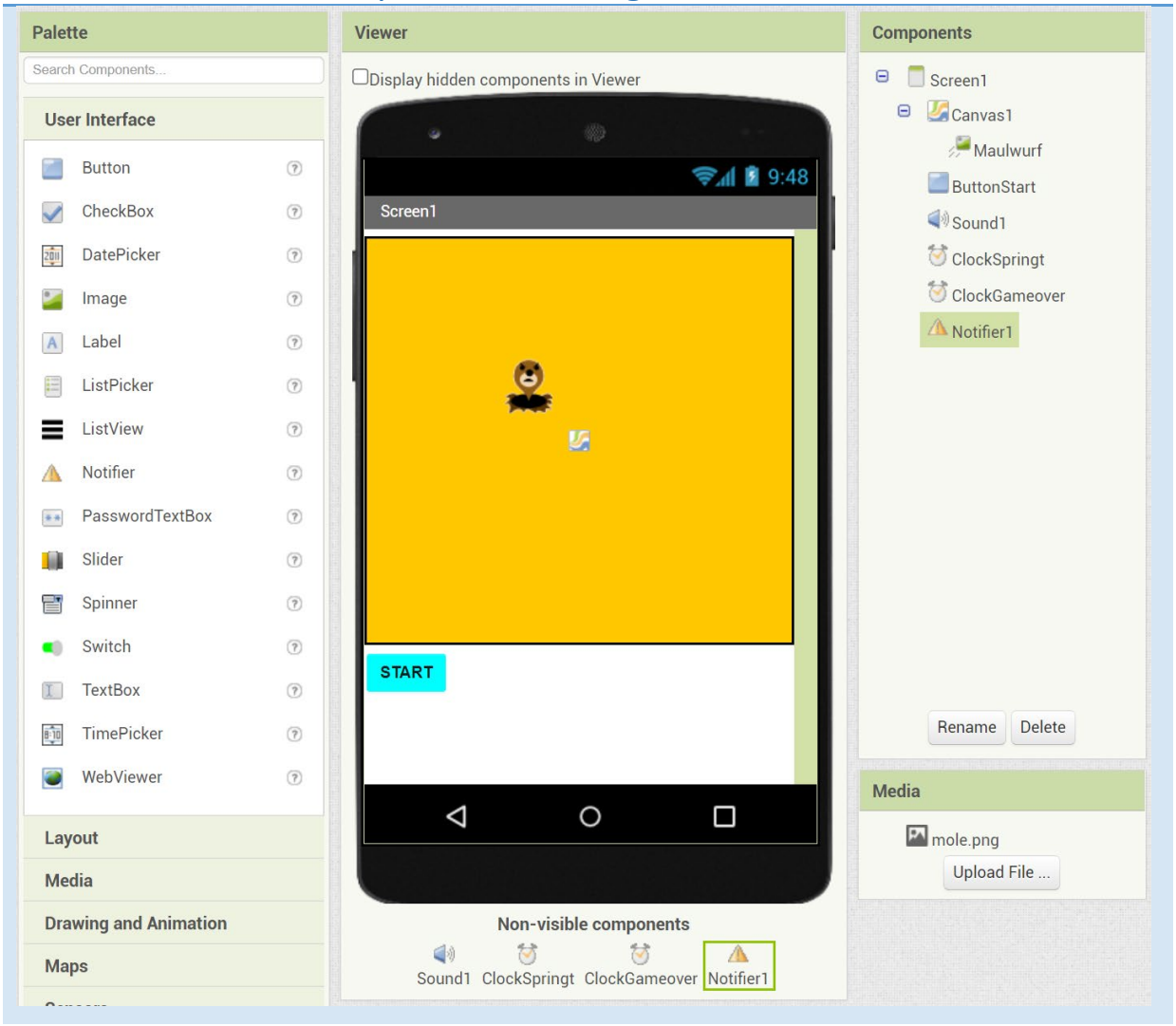
Text join:	*Hängt alle Eingaben zusammen, um eine einzige Textzeile zu bilden.*

Starte ein neues Projekt und nenne dieses MaulwurfSpiel_einfach. Füge die folgenden Komponenten im Viewer (Designer-Fenster) ein. Du musst jede Komponente umbenennen und gewisse Eigenschaften anpassen.

Komponenten	Palette (Kategorie)	Umbennenen in	Wofür plus Eigenschaften
Canvas	Drawing and Animation	Canvas1	Spielfläche *Eigenschaften:* -BackgroundColor: <Nach Geschmack> -Height: 300 Pixels -Width: fill parent
ImageSprite	Drawing and Animation	Maulwurf Bmk: platzieren innerhalb von Canvas	Spielfigur *Eigenschaften:* Picture: mole.png
Button	User Interface	ButtonStart	Anzeige «START» Man drückt diesen Knopf, um das Spiel zu starten
Sound	Media	Sound1	Vibriert, wenn der Maulwurf berührt wird
Clock	Sensors	ClockSpringt	Zählt ein Zeitintervall kontinuierlich *Eigenschaften:* -TimerAlwaysFires: ja -TimerEnabled: nein -TimerInterval: 500
Clock	Sensors	ClockGameover	Zählt ein Zeitintervall kontinuierlich *Eigenschaften:* -TimerAlwaysFires: ja -TimerEnabled: nein -TimerInterval: 10000
Notifier	User Interface	Notifier1	Zeigt die Trefferpunkte am Ende des Spiels.

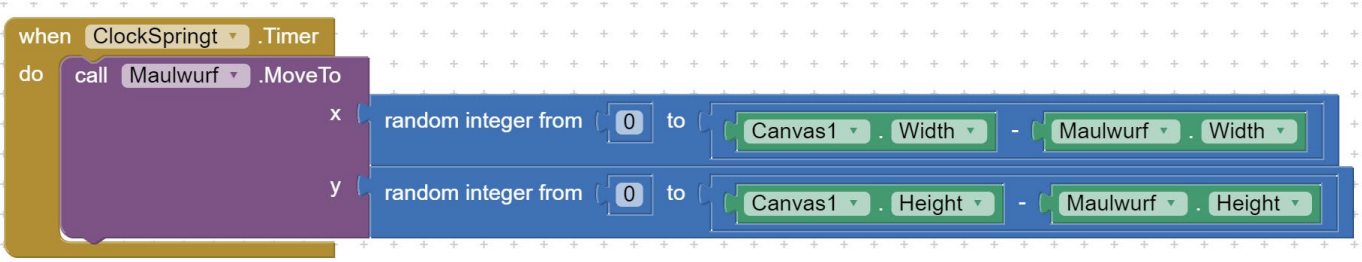
Überprüfe die Zusammenfassung des Designer-Teils unten. Hast du alles programmiert?

Übersicht Maulwurf-Spiel: Designer-Fenster



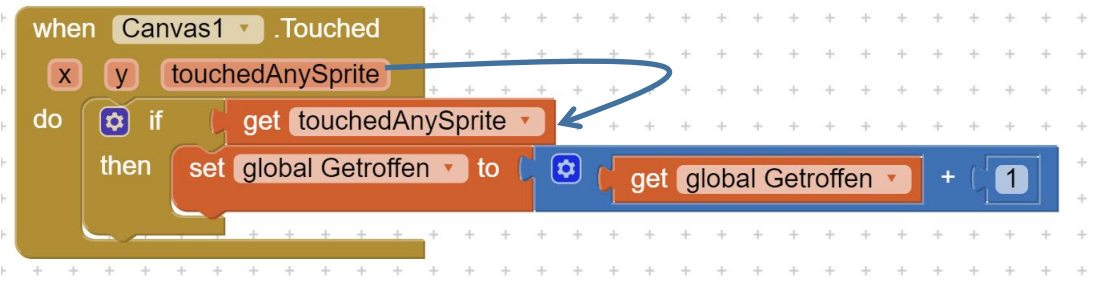
Jetzt müssen wir die entsprechenden Aktionen im Blocks-Fenster programmieren. Denk daran, deine App immer wieder zu testen.

1. Wir beginnen damit, das Hüpfen des Maulwurfs zu programmieren. Er soll nach jedem Intervall von ClockSpringt zufällig innerhalb des Canvas springen. Schau auf der linken Seite unter ClockSpringt, Maulwurf und Math nach den benötigten Aktionen und baue sie wie unten gezeigt. Wie soll die Programmierung sicherstellen, dass der Maulwurf innerhalb des Canvas bleibt? Diskutiere mit einer Nachbarin oder einem Nachbar.

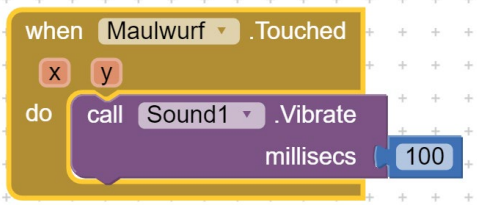


2. Wir brauchen auch einen Variablen, um die "Trefferpunkte" (Getroffen) zu zählen. Am Anfang hat man null Punkte.

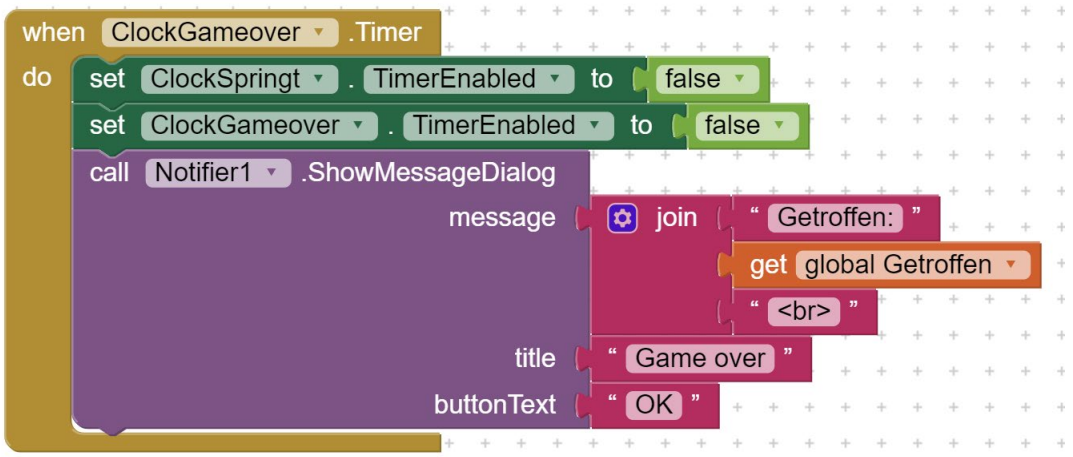
3. In den Funktionen von Canvas (Spielfläche) findest du einen Kontrollblock, der aktiviert wird, wenn der Spieler den Bildschirm berührt. Da kann man überprüfen, ob der Spieler den Maulwurf erwischt hat, und dann die Trefferpunkte erhöhen.



4. Wenn die SpielerIn den Maulwurf fängt, soll das Telefon vibrieren. Füge die passende Aktion hinzu.



5. Am Ende der Spielzeit (ClockGameover) wollen wir alle Clocks anhalten und die Gesamttrefferpunkte mit einem Notifier anzeigen. Finde die Aktionen unter den entsprechenden Blöcken, und baue sie wie unten gezeigt. Finde heraus, was der String
 im Text Join Block ändert.

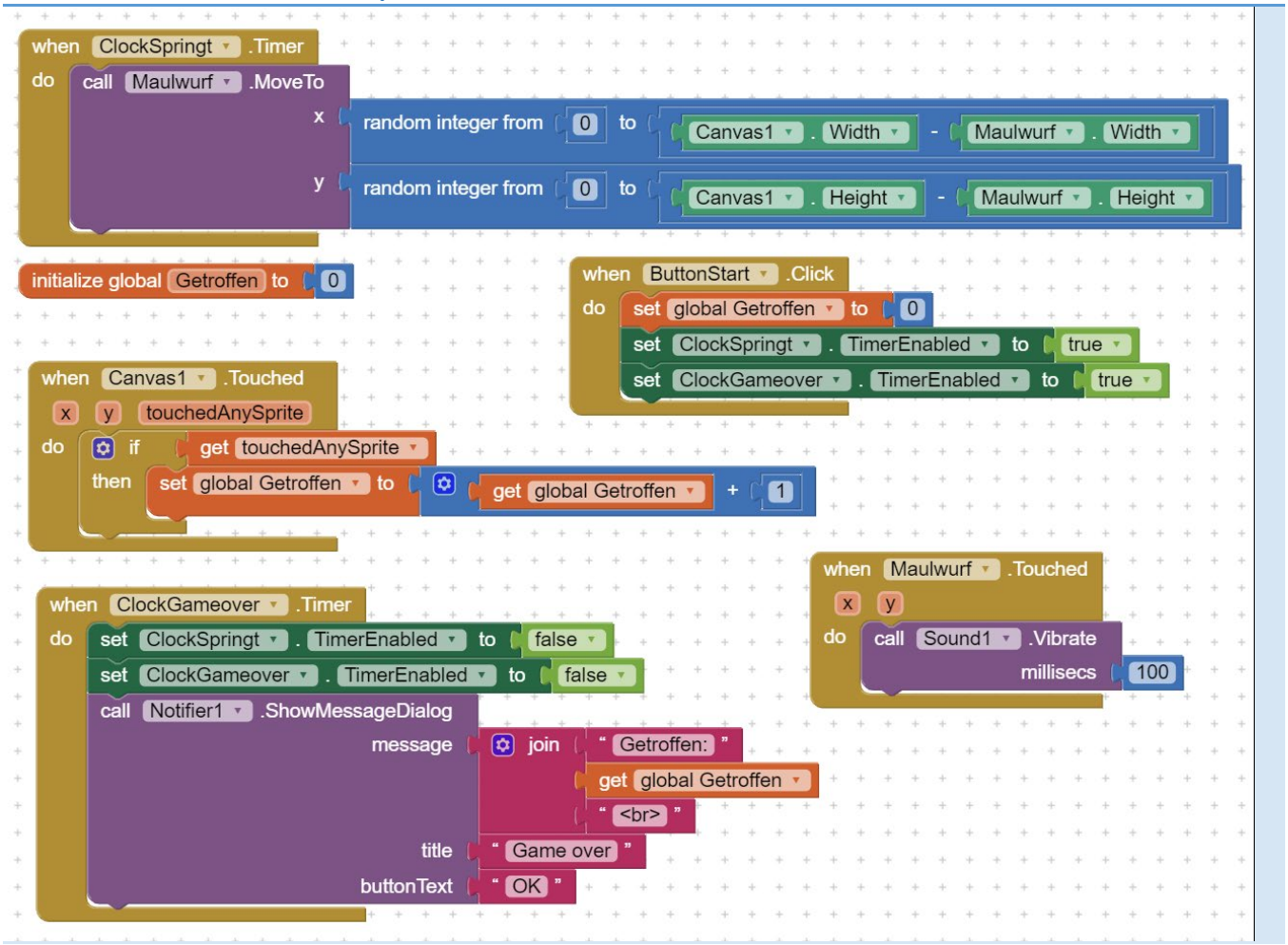


6. Denk jetzt darüber nach, wie man das Spiel starten und immer wieder neu starten kann. Sicher muss man die Trefferpunkte zurücksetzen, und die Clocks einschalten (TimerEnabled auf True setzen, siehe Snapshot unten).

Den Startbutton hast du bereits in dem Designer eingebaut. Programmiere jetzt alle Aktionen, die du brauchst!

Alles parat? Zeit zum Testen und Korrigieren. Überprüfe deinen Code mit der Übersicht auf der nächsten Seite.

Übersicht Maulwurf-Spiel: Blocks-Fenster



HERAUSFORDERUNGEN :

(1) Baue eine zweite Variable, Verpasst, und zähle auch die verpassten Punkte (wenn man den Maulwurf beim Drücken des Bildschirms nicht erwischt). Zeige am Ende des Spiels sowohl "Getroffen" als auch "Verpasst" Punkte mit dem Notifier an.

(2) Erweitern Sie das Spiel, sodass das Intervall zwischen zwei Sprüngen je nach Leistung des Spielers variiert. Zum Beispiel: kürzer, wenn es mehr Treffer als Fehler gibt, und umgekehrt.

(3) Und noch weitere Ideen von dir....

Die originelle detaillierte Anweisung für das «Maulwurf-Spiel» Tutorial findet man unten:

http://appinventor.mit.edu/explore/ai2/molemash.html

Oder runterladen das Kapitel:

http://www.appinventor.org/apps/molemash/molemash.pdf

BEMERKUNG
Das Original-Maulwurf-Tutorial verwendet Beschriftungen (Labels), Tabellenanordnungen (Table Arrangements) und Prozeduren (Procedures), die sicherlich interessante Ressourcen sind, aber insgesamt länger zum Programmieren brauchen. Diese vereinfachte Version ist gedacht für kurze Workshops.

ANHANG

Lösung zur Herausforderung (1)