Farbcodierung

Die App – Anleitung

Man hat zwei Möglichkeiten: Entweder man gibt in den Eingabefeldern für die Farben Rot, Grün und Blau (RGB) eine Zahl im Dezimalsystem zwischen 0 und 255 ein, lässt dann durch Klick auf den Button mit der Aufschrift „Färben“ dessen Hintergrundfarbe entsprechend ändern und den Hexadezimalcode der Farbe anzeigen.

Im unteren Bereich der App kann man eine zufällige Farbe erzeugen lassen, die den Hintergrund färbt. Mit den Schiebereglern lässt sich die Farbe des vorher hellgrauen Buttons unter den Schiebereglern verändern.

Ziel ist es, die Farbe des Buttons der Hintergrundfarbe anzugleichen. Vermutet man, die richtige Farbe gefunden zu haben, drückt man auf den Button mit der Aufschrift „Tipp überprüfen“ und bekommt dann Hinweise, ob die Farben stimmen oder ob noch zu viel oder zu wenig einer Farbe vorhanden ist.

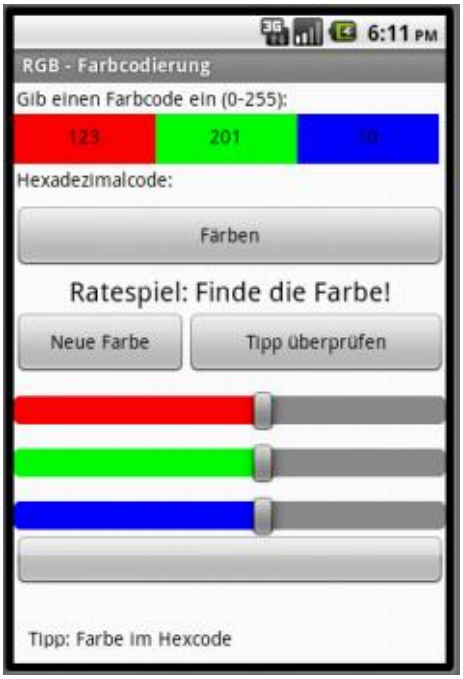
Da die Schieberegler nicht ganz einfach schrittweise verändert werden können, kann man auch oben in den Eingabefeldern die Werte verändern und durch Drücken der Schaltfläche „Färben“ dem Spiel übergeben. Erneutes Drücken von „Tipp überprüfen“ gibt an, wie es aktuell aussieht.

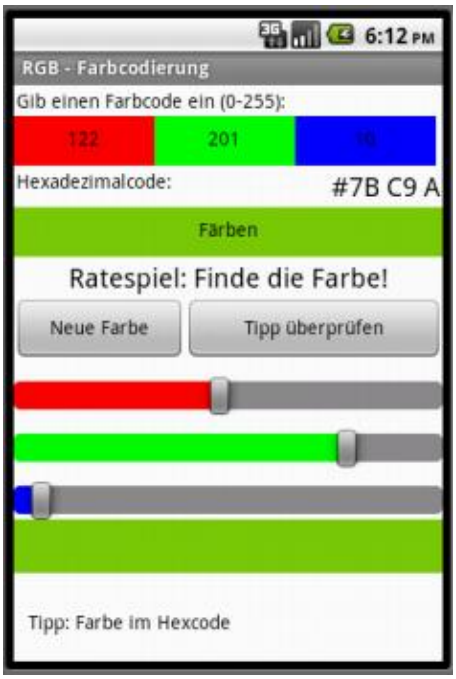
Ganz unten im Bildschirm gibt es einen Hilfebutton. Drückt man darauf, bekommt man den Hexcode der gesuchten Farbe angezeigt. Vergleicht man diesen mit dem Hexcode oben, kann man auch – wenn man sich mit Hexadezimalzahlen etwas auskennt – die gesuchten Werte im Dezimalsystem in den Feldern oben eingeben[1].

Der Screen

In der ersten Abbildung sieht man den Startbildschirm der App. Nach dem Klick auf „Färben“ sieht man den Screen der zweiten Abbildung.

Oben wird der Hexcode zu den Dezimalzahlen angezeigt.

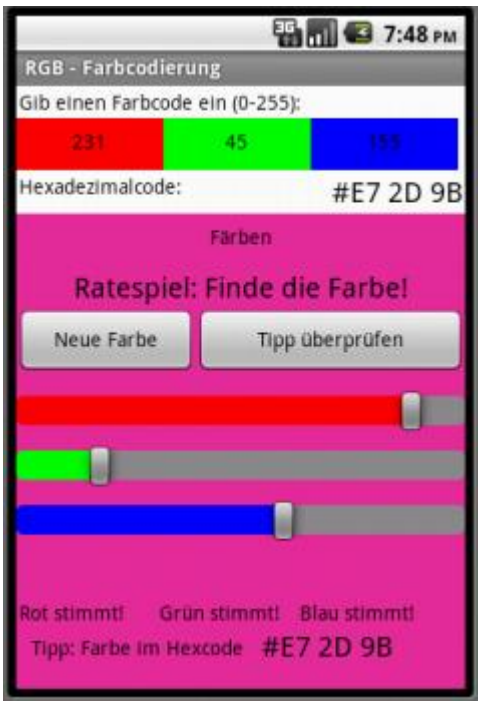




1 Bei der ersten Eingabe muss man manchmal noch korrigieren. Da hat sich noch ein Bug versteckt.

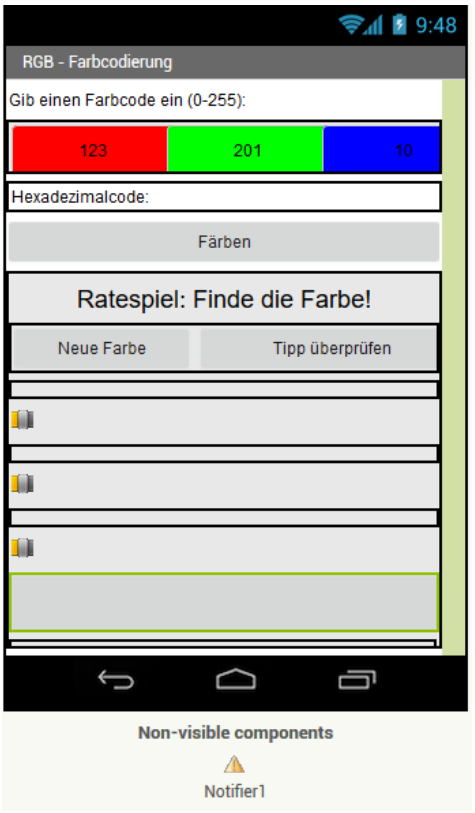
In der dritten Abbildung wurde eine neue Farbe erzeugt. In der vierten wurden die Schieberegler bewegt und der Tipp wurde überprüft.

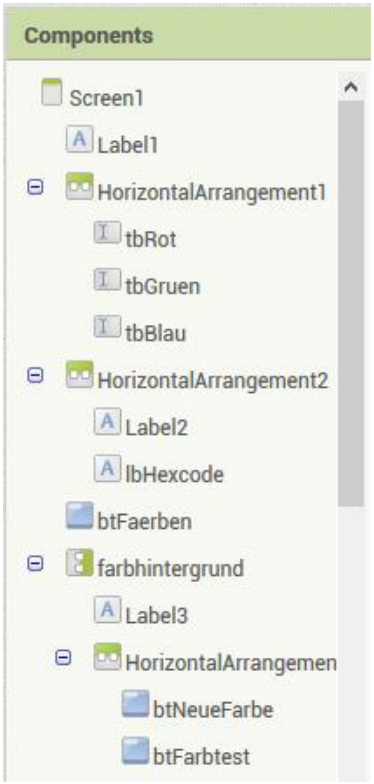
In der fünften Abbildung ist die Farbe fast richtig. Das sieht man am Hexcode, der unten nach dem Drücken des Tipp-Buttons angezeigt wird. In der letzten Abbildung stimmt die Farbe.

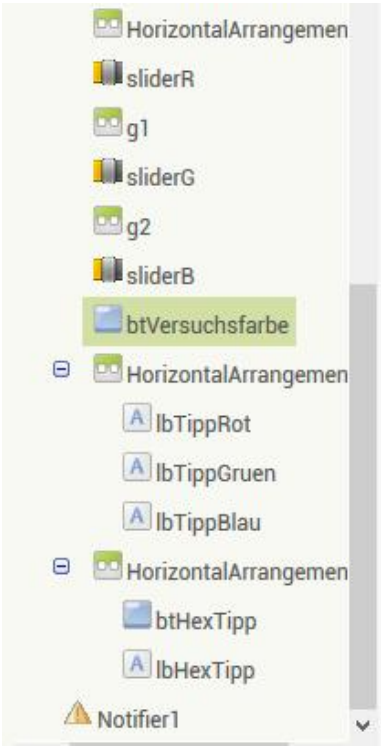


Der Screen im MIT-App-Inventor

Ein Teil der Komponenten ist nicht mehr sichtbar im unteren Bereich.







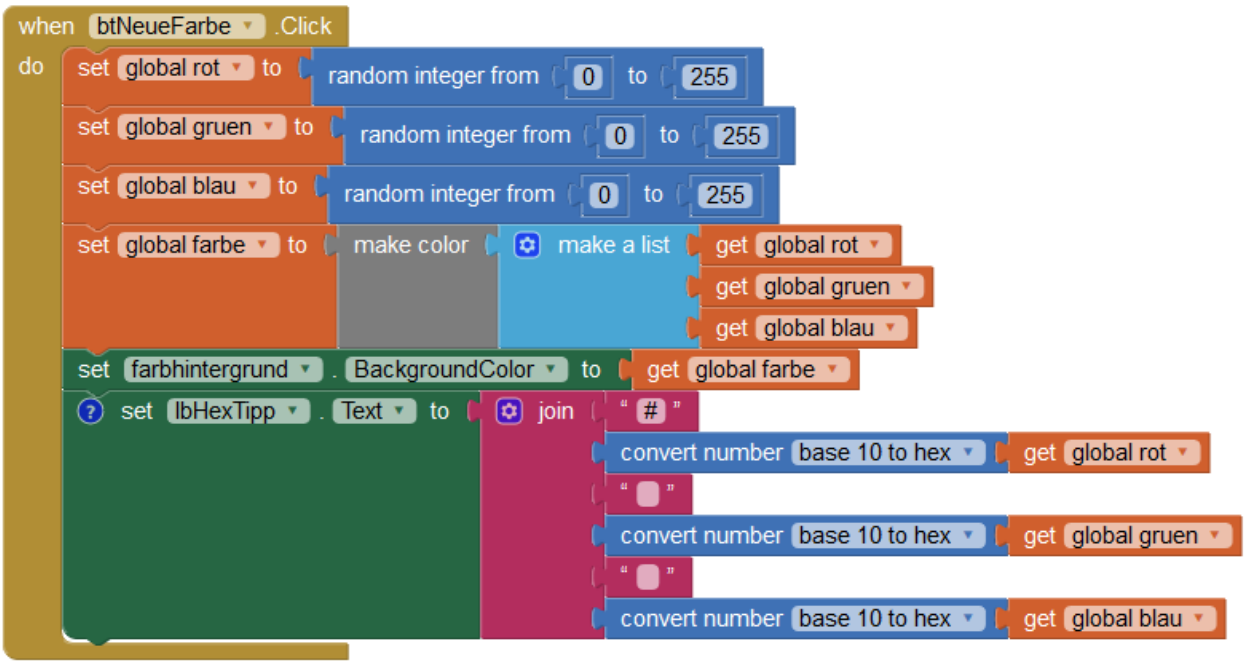
Komponenten, die lediglich Information geben aber im Code nicht verwendet werden, wurden nicht umbenannt.

Der Code

Initialisierung der Variablen:

Es gibt eine Variable für die gesuchte Farbe und jeweils eine weitere Variable für den Anteil an rot, grün und blau der Farbe. Die *test*-Variablen sind dazu da, Zwischenwerte zu speichern, um mit den gesuchten Werten vergleichen zu können.

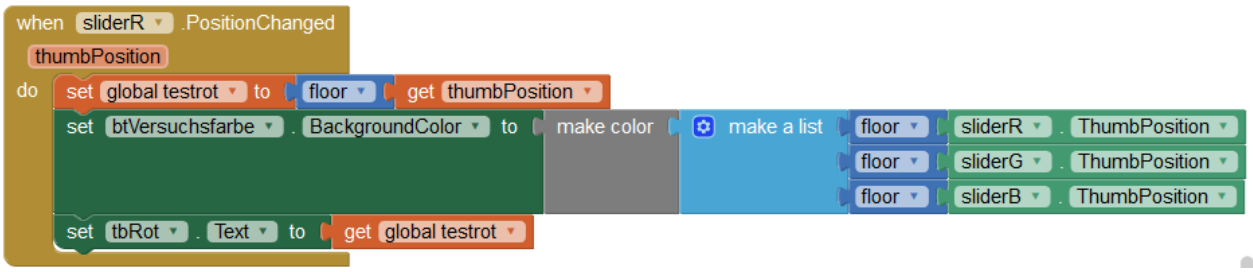
Das OnClick-Ereignis des Buttons mit der Aufschrift „Neue Farbe“:



Es werden drei Zufallszahlen zwischen 0 und 255 erzeugt und in den Variablen für die Farbanteile gespeichert. Diese Werte werden als Liste mit drei Elementen in eine Farbe umgewandelt und in der Variable *farbe* gespeichert.

Der Hintergrund wird gefärbt und die Werte werden in den Hexadezimalcode umgewandelt. Angezeigt wird dieser erst, wenn der entsprechende Button gedrückt wurde. Solange ist das Label unsichtbar.

Die OnPositionChanged-Ereignisse der Schieberegler (Slider) – exemplarisch für den ersten der drei:



Da die Position der Schieberegler keine ganze Zahl ist, wird hier durch „floor“ die nächstkleinere ganze Zahl zunächst in der *test*-Farbvariable gespeichert. Die Farbe, die aus allen drei Farbwerten zusammengesetzt ist, wird als Hintergrundfarbe des Buttons unter den Schiebereglern verwendet.

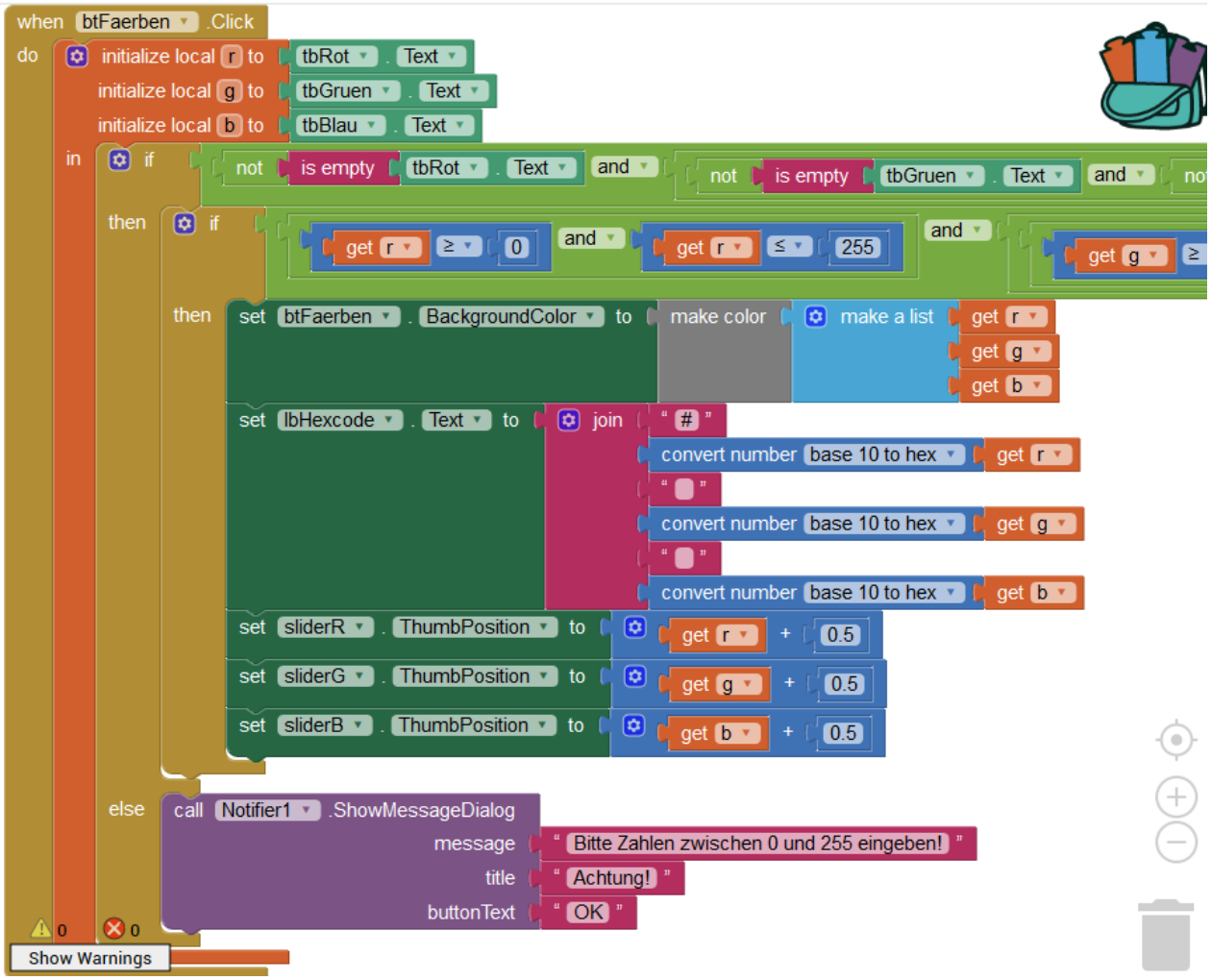
Der Wert der Variable *testrot* (analog für grün und blau) wird in das Textfeld oben geschrieben.

Das OnClick-Ereignis des Buttons „Tipp überprüfen“:

Alle drei Farbanteile werden verglichen und die entsprechende Meldung ausgegeben.

Zum Anzeigenlassen des Hexcodes:

Das OnClick-Ereignis des Buttons „Färben“:



In drei lokalen Variablen werden die Zahlen aus den Textfeldern für die drei Farbanteile gespeichert.

In der ersten Verzweigung wird zunächst überprüft, dass keines der Felder leer ist. Wenn das doch der Fall sein sollte, wird eine Nachricht ausgegeben. Die zugehörige Komponente ist ein sogenannter Notifier.

Weiter wird überprüft, ob alle drei Werte im richtigen Bereich zwischen 0 und 255 liegen. Wenn ja, wird der Button selbst gefärbt, der Hexcode angegeben und die Schieberegler auf die Position gesetzt.

Bemerkungen

Es gibt keine Fehlerabfrage, sollte eine Dezimalzahl eingegeben werden.

Den Code zum Bearbeiten findet man unter *Farbcode.aia*, die App zum Installieren unter *Farbcode.apk*.

Der MIT App Inventor (*http://appinventor.mit.edu*) wurde ursprünglich von einem Entwicklerteam um Mark Friedman und Hal Abelson bei Google entwickelt und 2012 an das MIT übergeben.

Scratch wird von der Lifelong-Kindergarten-Group am MIT-Media-Lab entwickelt. Siehe *http://scratch.mit.edu*.