

# tac tile

*Sebastian Bast, Anton Krapp, Melissa Zaczyk und David Junk*

## Ziel und Motivation

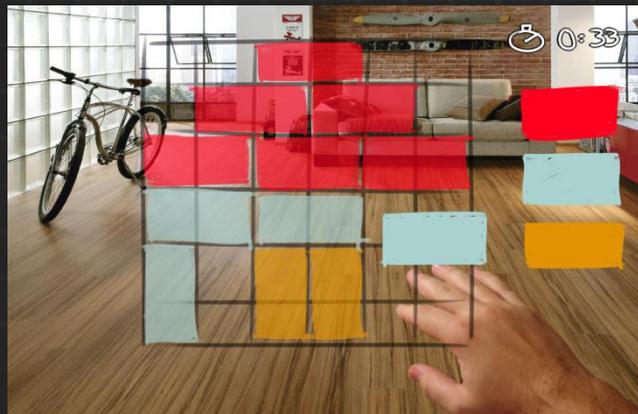
Ziel des Projekts ist ein Prototyp eines AR Spiel für Android zu entwickeln. Für die Steuerung des Spiels sollen ausschließlich Gesten genutzt werden. Die Gesten sollen möglich einfach, intuitiv und für den Spieler klar von einander zu unterscheiden sein. Unsere Idee war, ein Spiel umzusetzen, bei dem virtuelle Objekte im Raum mithilfe einer Gestensteuerung manipuliert werden können. Im weiteren Verlauf kam uns die Idee, ein Spiel mit Anlehnung an das Brettspiel „Ubongo“ zu entwickeln. Der Spieler soll durch das Manipulieren von virtuellen Steinen z.B. Auswählen, Verschieben, Drehen und Platzieren bestimmte Grundformen und Figuren in einer vorgeschriebenen Zeit nachbauen.

## Spielidee

In einer vorgegebenen Zeit soll der Spieler Motive aus unterschiedlichen Formen und Farben nachbauen. Danach sollen diese Motive nach dem Gedächtnis nachgelegt werden. Das heißt, dass der Spieler sich die Motive merken muss. Die maximale Größe der Motive ist auf ein Gitter aus 7x7 Feldern begrenzt. Felder die nicht genutzt werden sollen, werden für den Spieler ausgeblendet. Die Bausteine decken 1-3 Felder ab.

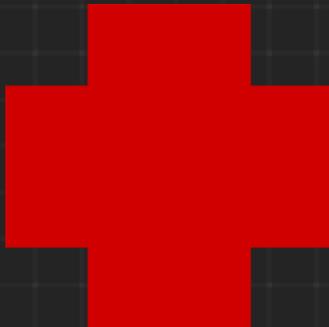
## Ablauf

-  Zu Beginn wird das zu legende Motiv angezeigt und nach Ablauf einer Zeit wieder ausgeblendet.
-  Der Spieler muss sich das Motiv in der Zeit merken.
-  Anschließend muss der Spieler das Motiv in der vergebenen Zeit mit der Gestensteuerung nachlegen.
-  Nur die zu besetzenden Felder werden dem Spieler angezeigt.
-  Ist das Motiv korrekt, so erreicht der Spieler das nächste Level.
-  Ist es falsch, dann wird das Motiv wiederholt.
-  Die Motive werden mit jedem Level komplexer und somit schwerer.

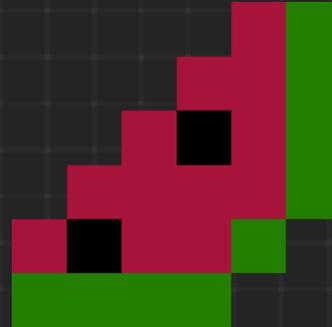


## Bausteine

-  Quadrate
-  Rechtecke
-  Weitere Steine zusammengesetzt aus Quadraten
-  Variation in den Motiven durch verschiedene Farben



Kreuz



Melone

## Verwendete Technologien

Als Entwicklungsplattform wurde für die App Unity verwendet. Da sich AR Anwendungen und Spiele mit der Plattform gut umsetzen lassen. Wir haben in unserem Projekt die aktuellste Unity Version 2019.3.13f1 Personal genutzt. Zusätzlich nutzten wir das Framework AR Foundation. AR Foundation bietet die Anbindung an ARCore von Google, die Programmierschnittstelle für AR-Anwendungen für Android-Geräte. Für die Erkennung und Erstellung der Handgesten haben wir das Framework ManoMotion SDK Lite genutzt.

## Gesten

### Grundgesten

In ManoMotion gibt es drei Gestenfamilien, aus denen die Handgesten erstellt werden können:

-  Pinch
-  Grab
-  Pointer



## Pick / Auswählen des Bausteins

Daumen und Zeigefinger werden über dem Baustein zusammengeführt. Anschließend heftet der Stein an der Position der Hand.

## Verschieben des Bausteins

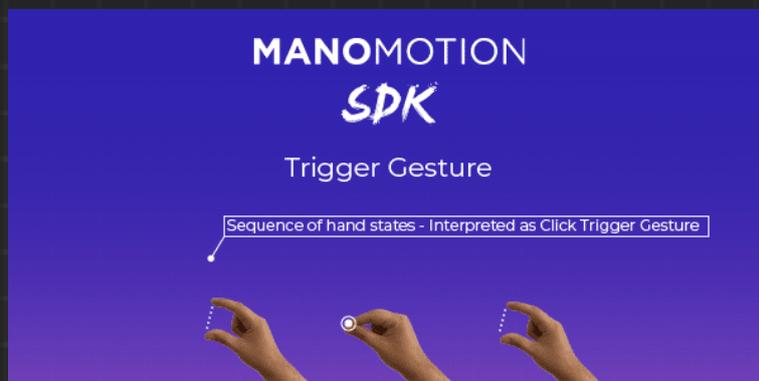
Wenn der Baustein erfolgreich ausgewählt wurde, kann durch Verschieben der Hand der Baustein bewegt werden. Dieser folgt dann der Bewegung der Hand.

## Drop / Ablegen des Bausteins

Daumen und Zeigefinger werden über dem Platz im Gitter, wo der Stein abgelegt werden soll, wieder geöffnet. Der Baustein rastet danach an seinem Platz ein.

## Trigger / Drehen des Steins

Daumen und Zeigefinger werden über dem Stein einmal zusammengeführt und direkt wieder geöffnet. Pro Aktion (Klick) wird der Stein jeweils um 90° gedreht.



## Grab / Leeren des Grid

Die Finger werden über dem Grid zu einer Faust zusammengeführt. Anschließend wird die geschlossene Hand nach **rechts** bewegt **und wieder geöffnet**. Alle bisher gelegten Steine werden nun aus dem Grid entfernt.

## Hardwareunterstützung

ManoMotion und die Google Schnittstelle ARCore werden nur von den neueren Endgeräten unterstützt. Auf alten Geräten ist unsere App daher leider nicht lauffähig.

## Voraussetzungen:

-  **ARcore unterstützendes Endgerät:** zum Beispiel „Google Pixel“  
Alle Geräte zu finden unter: <https://developers.google.com/ar/discover/supported-devices>
-  **Betriebssystem:** API Level 24, Nougat, Version 7.0 oder höher
-  **CPU:** Octa-core (4×2.1 GHz Cortex-A57 & 4×1.5 GHz Cortex-A53) oder höher
-  **RAM:** 3GB
-  **Architektur:** Armeabi-v7a
-  **Internetverbindung:** Aktive WLAN- oder Mobilfunkverbindung

## Weitere Informationen

Im Gitlab /Wiki sind weitere detaillierte Informationen und eine Dokumentation über den Werdegang des Spieles zu finden <https://github.com/brokensmile-de/tactile>