

# Leitfaden: Programmieren der MBots

## Erste Schritte:

1. Das Programm „mBlock Blockly“ auf dem Gerät installieren
2. Den Roboter mit Bluetooth verbinden

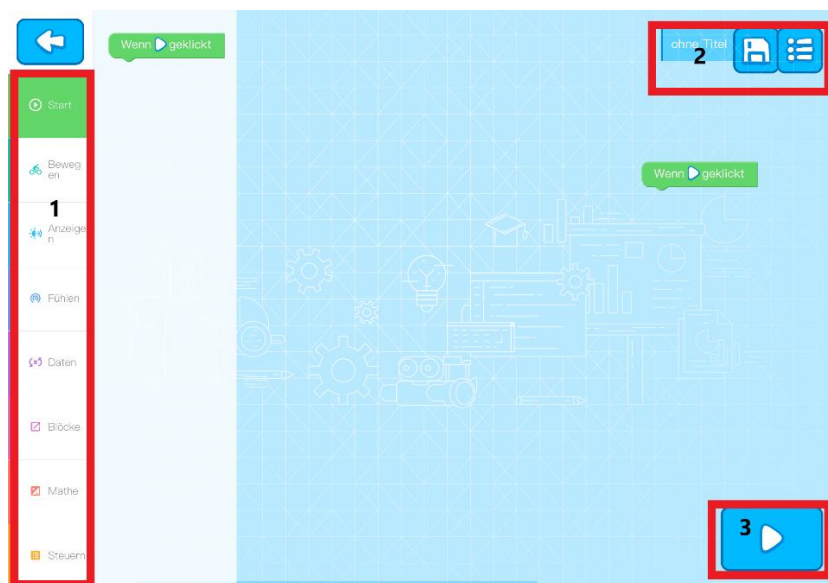
## Vorab Hinweis:

Das Programm lädt viel zum Trial & Error Programmieren ein. Besonders, wenn man einen Parkour baut, den der Roboter Schritt für Schritt bewältigen muss. Es kann hilfreich sein sich Schritt für Schritt durch die verschiedenen Bausteine zu probieren und deren Funktionen zu testen.

## Das Programm:

*Hinweis:* nachfolgende Bilder sind aus der Apple Version des Programmes aufgenommen, die Android App kann sich unter Umständen unterscheiden

## Bild 1: Übersicht:



### Abschnitt 1:

In diesem Abschnitt sind die verschiedenen **Programmbausteine** sortiert:

Bewegung, Anzeigen, Fühlen, Daten, Blöcke, Mathe und Steuern.

*(Weiter unten dazu mehr Details)*

### Abschnitt 2:

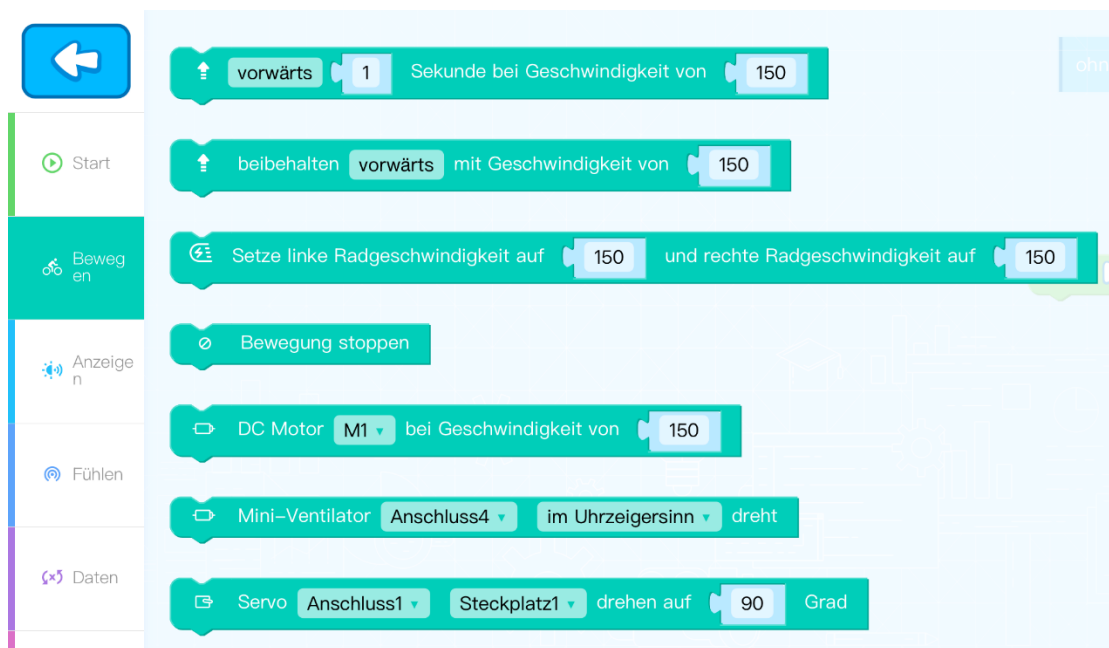
hier kann man das Projekt speichern und in die Einstellungen des Programmes gehen.

### Abschnitt 3:

drückt man diesen Button, wird der aktuelle Code vom verbundenen Roboter ausgeführt.

## Die Verschiedenen Bausteine:

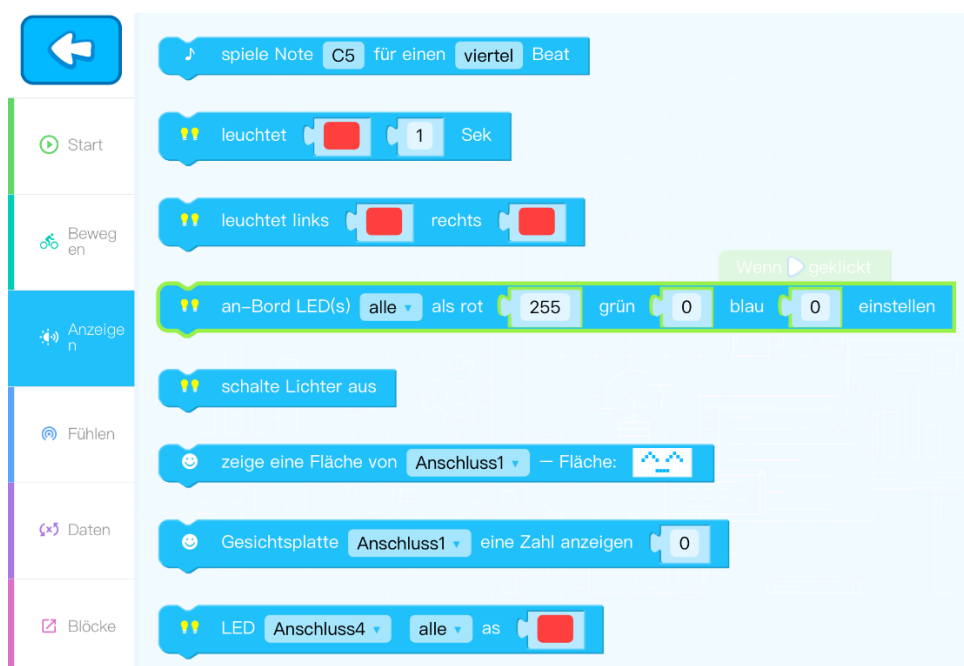
Bild 2: Bewegung



**Bewegung:** Hier sind verschiedene Bausteine um die Bewegung des Roboters zu steuern.

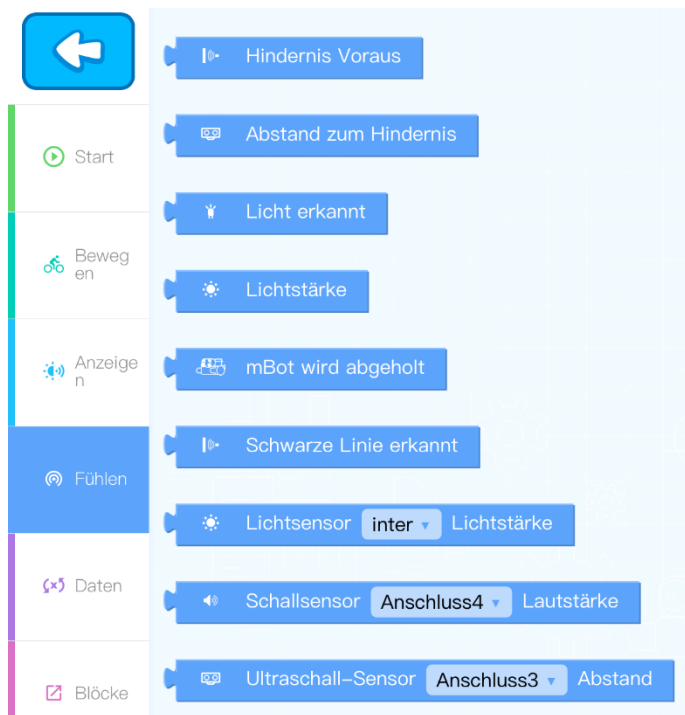
*Hinweis:* Man kann die Geschwindigkeit mit **Mathe-Blöcken** genauer anpassen. Bei einigen Bausteinen gibt es keine Zeitangabe. Hier helfen **Steuerungsblöcke** aus (Siehe Bild 8).

Bild 3: Anzeigen



**Anzeigen:** Verschiedene Bausteine, die mit dem eingebauten LED-Lichter des Robots und einem ggf. vorhandenen Bildschirm interagieren.

Bild 4: Fühlen

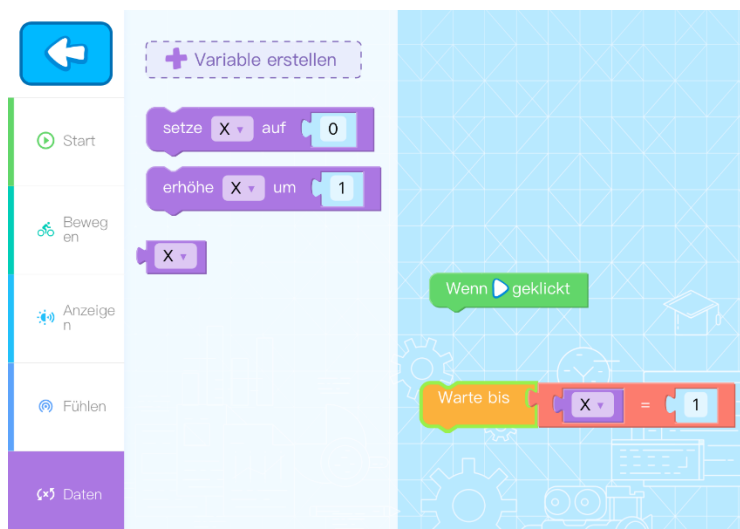


**Fühlen:** Hier sind die verschiedenen Sensoren des Roboters untergebracht

*Hinweis:* Wenn ein Sensor funktioniert, erhält man vom Programm oben rechts Feedback, zur aktuellen Messung des Sensors

*Notiz zu „Schwarze Linie erkannt“:* Hier wird ein wirklich heller Untergrund benötigt, von dem der Roboter die Schwarze Linie Eindeutig erkennen kann.

Bild 5: Daten



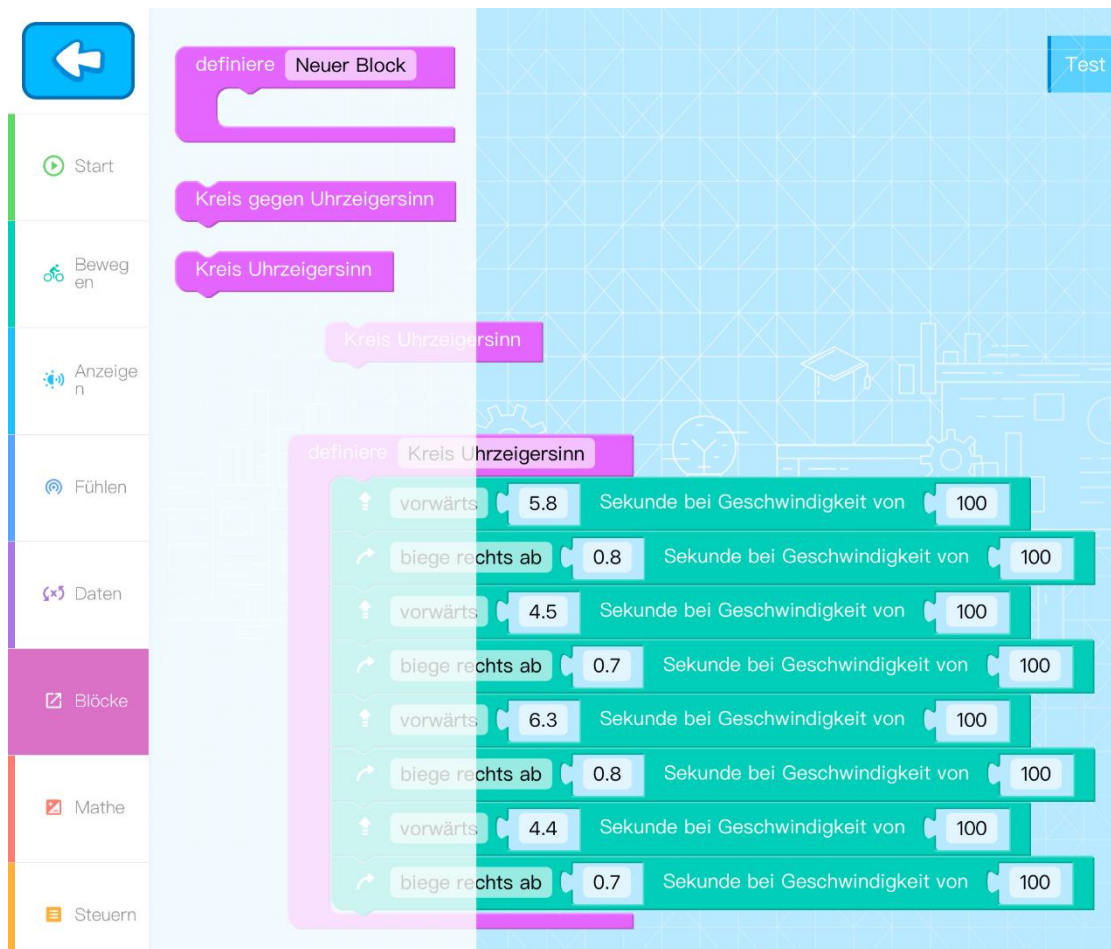
**Daten:** Hier kann man Variablen für komplexere Programmieraufgaben erstellen. Mit Variablen lassen sich Zahlen zählen oder verändern.

*Beispiel:* Man kann eine Variable mit dem Namen „X“ erstellen. Mit deren Hilfe kann man den Roboter warten lassen, bis er der Wert der Variable X einen bestimmten Wert hat. Sobald die Variable dann den Wert X

hat, wird der Roboter aufhören zu warten. Und das Programm läuft weiter.

*Hinweis:* Hierfür ist eine Kombination mit **Mathe-** und **Steuerungsbausteinen** notwendig.

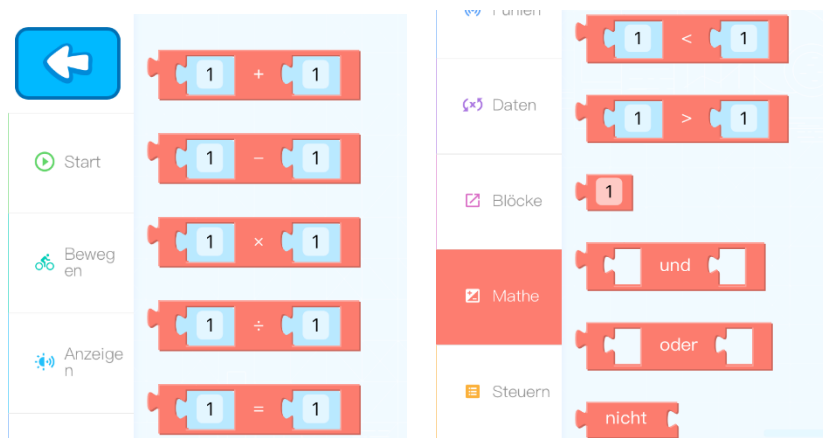
Bild 6: Blöcke



**Blöcke:** Hiermit kann man sich wiederholende Befehlsketten in einzelne Bausteine umwandeln.

*Beispiel:* Auf dem Bild ist zu sehen, wie ein Baustein programmiert wurde, durch den der Roboter eine bestimmte Route im Uhrzeigersinn abfahren soll. Mithilfe des „Blocks“ kann man solche einzelnen Bausteine vor Definieren und dann einfach in die Befehlskette einbauen.

Bild 7: Mathe

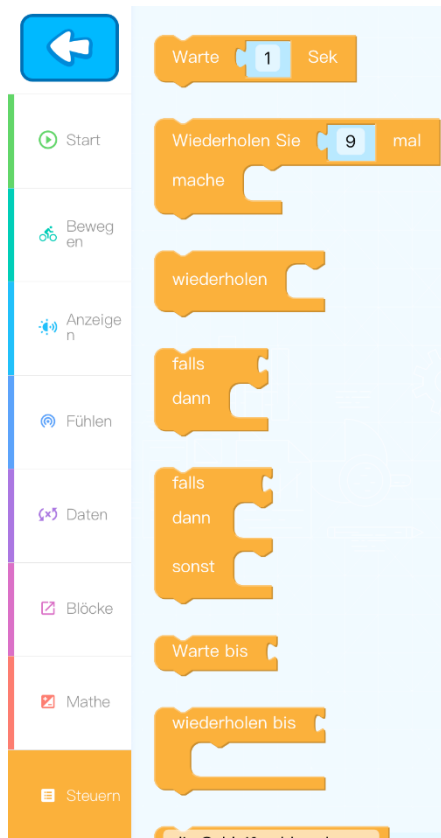


**Mathe:** Hier kann man diverse Mathematische Funktionen mit in die Programmbausteine einbauen

*Hinweis:* Wie bereits weiter oben beschrieben, ergeben sich hierdurch weitere komplexe Möglichkeiten zum

Programmieren.

Bild 8: Steuern:



**Steuern:** Oftmals ist eine Kombination mit diesen Bausteinen notwendig, um andere entsprechend funktionieren zu lassen.

*Als Beispiel, siehe die Kombination von Blöcken weiter unten.*

Mit den Steuerbausteinen kann man auch komplexere Programmierungen erstellen. Ein Sinnvoller Einsatz von **Sensoren** erfordert meistens eine tiefere Auseinandersetzung mit dieser Art von Bausteinen.

Bild 9: Beispiel Kombination von Blöcken

Hier Ein Beispiel von der **Kombination verschiedener Blöcke**. Die Blöcke sind ein wenig wie **Puzzleteile** geformt und können ineinander eingesetzt werden.

Im *Beispiel* sieht man eine Programmierung, bei der die Drehgeschwindigkeit der Räder über die Standardeinstellungen (Langsam, Mittel, Schnell) hinaus angepasst wurden. Außerdem wurden **Steuerungsbausteine** „Warten“ Eingefügt, damit der Roboter die gewünschte Bewegung für einen bestimmten Zeitraum ausführt.

