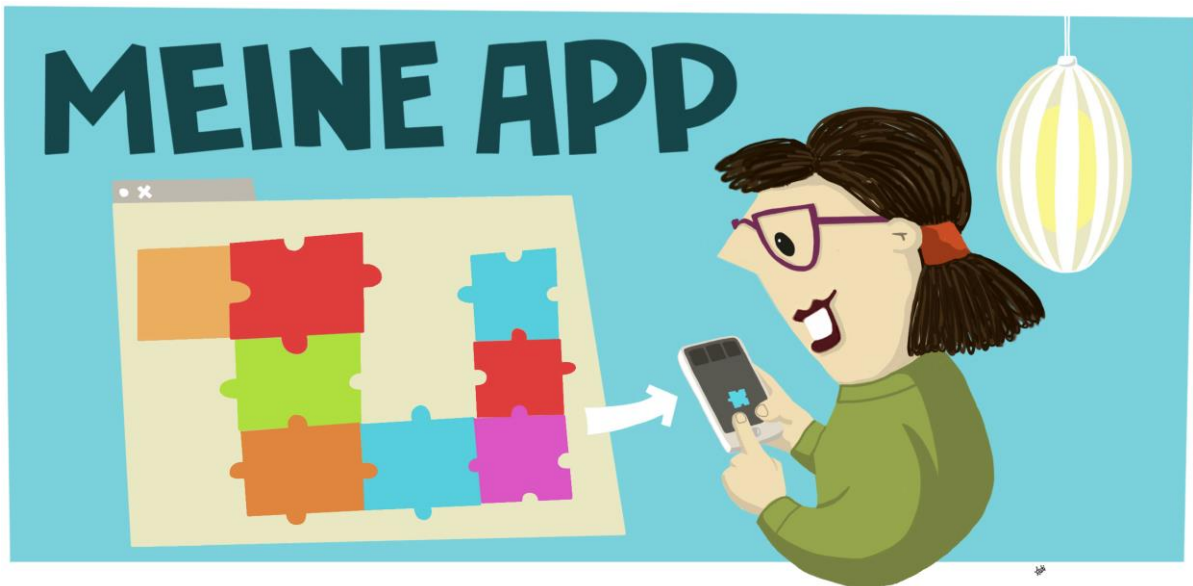


# IT2School

Gemeinsam IT entdecken



## Modul B7 – Meine App

App Inventor

Eine Entwicklung von



In Kooperation mit



Im Auftrag der

# Inhalt

1	App Inventor.....	3
2	Warum gibt es das Modul? .....	4
3	Ziele des Moduls.....	4
4	Die Rolle der Unternehmensvertreterin/des Unternehmensvertreters.....	4
5	Inhalte des Moduls.....	5
5.1	Aufruf und Einstieg in den App Inventor .....	5
5.2	Die grafische Benutzeroberfläche.....	5
5.2.1	Entwicklungsumgebung Design Editor.....	6
5.2.2	Entwicklungsumgebung Blocks Editor .....	7
6	Unterrichtliche Umsetzung.....	7
6.1	Grober Unterrichtsplan.....	8
6.2	Stundenverlaufsskizzen .....	8
6.2.1	Einführung, Kennenlernen und Entwicklung einer App .....	8
6.2.2	Vertiefung: Eigene Projekte.....	8
7	Einbettung in verschiedene Fächer und Themen .....	8
8	Anschlussthemen.....	9
9	Literatur und Links .....	9
10	Arbeitsmaterialien .....	10
11	Glossar.....	10

# 1 App Inventor

Dieses Modul dient der Vertiefung im Bereich der Programmierung und Anwendungsentwicklung auf Android-Smartphones. Dabei bietet es viel Spielraum für die Kreativität der Schülerinnen und Schüler, aber auch der Lehrkraft selbst. Als Werkzeug für die Entwicklung von Applikationen (kurz: Apps) auf dem Smartphone dient der ursprünglich von Google entwickelte *App Inventor*, eine bausteinartige Entwicklungsumgebung, die sich ähnlich wie Scratch verhält.



Die Verbreitung und Nutzung von Smartphones nimmt unter Schülerinnen und Schülern (auch in der Sek. I) seit Jahren stark zu. Die Entwicklung eigener Apps ist für die Schülerinnen und Schüler daher ein naheliegendes Vorhaben und bedarf keiner gesonderten Motivation. Die Entwicklung selbst ist nicht nur Spielerei, sondern kann auch für den Unterricht und die Lehrkraft von Interesse sein. In verschiedenen Fächern lassen sich Themenfelder finden, für die das Smartphone als Werkzeug programmiert und genutzt werden kann.

<b>Lernfeld/Cluster:</b>	IT spielend entdecken	
<b>Zielgruppe/Klassenstufe:</b>		4. bis 5. Klasse
	X	6. bis 7. Klasse
	X	8. bis 10. Klasse
	X	11. bis 12. Klasse
<b>Geschätzter Zeitaufwand:</b>	4 bis 6 Einzelstunden	
<b>Lernziele:</b>	<ul style="list-style-type: none"> <li>• ein grundlegendes Programmierverständnis entwickeln</li> <li>• Eigenschaften von Programmen beschreiben</li> <li>• erleben wie leicht kleine Programme selbst gestaltet werden können</li> <li>• algorithmische Grundbausteine zur Programmierung verwenden</li> <li>• Programme entwerfen, implementieren und testen</li> <li>• Probleme selbstständig lösen</li> </ul>	
<b>Vorkenntnisse der Schülerinnen und Schüler:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Erfahrung mit Scratch oder anderen Programmiersprachen</li> </ul>	
<b>Vorkenntnisse der/des Lehrenden:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Erfahrung in der Programmierung</li> </ul>	
<b>Vorkenntnisse der Unternehmensvertreterin/des Unternehmensvertreters:</b>	Empfohlen: <ul style="list-style-type: none"> <li>• Erfahrung in der Programmierung</li> </ul>	
<b>Sonstige Voraussetzungen:</b>	Erforderlich: <ul style="list-style-type: none"> <li>• genügend Computer mit einer stabilen Internetverbindung (App Inventor kann nicht offline ausgeführt werden)</li> <li>• ein anderer Browser als der Internet Explorer (da dieser vom App Inventor nicht unterstützt wird)</li> </ul>	

## 2 Warum gibt es das Modul?

Ähnlich wie in den Modulen B4 und B6 wird in diesem Modul den Schülerinnen und Schülern Raum für eigenständiges Entwickeln und Austesten gegeben. Der Fokus liegt dabei auf einer Vertiefung der Programmierfähigkeiten, zugleich geht es darum, den Umgang mit anderen Informatiksystemen als dem Computer kennenzulernen.

Die Schülerinnen und Schüler werden für die Entwicklung von Anwendungen begeistert und erfahren, welche Einsatzmöglichkeiten das Smartphones (noch) bietet. Dabei entwickeln sie ein Verständnis für die Grundlagen der App-Entwicklung und können gezielt Problemlösungen und eigene Projekte angehen. In der aktiven Auseinandersetzung und Entwicklung eigener Projekte erfahren die Schülerinnen und Schüler mehr über moderne Arbeitsprozesse und lernen, dass es nicht zuletzt an ihnen liegt, die digitale Gesellschaft mitzugestalten.

Auch wenn die Heranführung an die Programmierung im Vordergrund steht, kann das Modul im zweiten Schritt sehr gut dazu verwendet werden, IT-Werkzeuge für Unterrichtsfächer außerhalb der Informatik zu entwickeln. So können die Schülerinnen und Schüler zum Beispiel eine Vokabel-App für den Fremdsprachenunterricht entwickeln.

## 3 Ziele des Moduls

- Ein grundlegendes Programmierverständnis entwickeln
- Erleben, wie leicht Apps für Android-Smartphones gestaltet werden können
- Softwareerstellung als kreativen Prozess entdecken
- Eigenschaften von Programmen analysieren und beschreiben
- Algorithmische Grundbausteine zur Programmierung verwenden
- Entwerfen, Implementieren und Testen eigener Programme

## 4 Die Rolle der Unternehmensvertreterin/des Unternehmensvertreters

Im Modul B7 – Meine App hat die Unternehmensvertreterin/der Unternehmensvertreter mehrere Möglichkeiten, aktiv mitzuwirken. Hier einige Anregungen:

- Unterstützung der Lehrkraft beispielsweise bei der Einführung in die Programmier-Oberfläche des App Inventors
- Unterstützung der Schülerinnen und Schüler bei der Umsetzung ihrer eigenen kreativen Projekte
- Sie oder er kann zur Abschlusspräsentation der Projektergebnisse als Special-Guest eingeladen werden.

## 5 Inhalte des Moduls

Zur Entwicklung von Anwendungen (engl.: *applications*, kurz: Apps) wird in diesem Modul das Programm App Inventor verwendet. Es handelt sich dabei um eine Web-Anwendung, die nur über den Browser und mit bestehender Internetverbindung aufgerufen werden kann.

App Inventor wurde von Google entwickelt und ermöglicht es, Anwendungen für das Android-Betriebssystem zu erstellen.

Die Bedienung des Programms ist vergleichbar mit Scratch – so wird für die Programmierung ebenfalls eine grafische Benutzeroberfläche genutzt, auf der per Drag-and-Drop Bausteine zu einem Skript kombiniert werden. Um die selbstgeschriebenen Apps zu testen, ist kein Android-Smartphone notwendig, da auf einen Emulator zurückgegriffen werden kann. Aber natürlich lassen sich die Apps auch auf dem Smartphone installieren. Eine wichtige Voraussetzung für die Nutzung des Programms ist neben der ständigen Internetverbindung ein Google-Nutzerkonto (über das Nutzer eines Android-Smartphones normalerweise verfügen oder das schnell erstellt werden kann). Für den Unterricht kann die Lehrkraft auch selbstständig einige Nutzerkonten anlegen.

### 5.1 Aufruf und Einstieg in den App Inventor

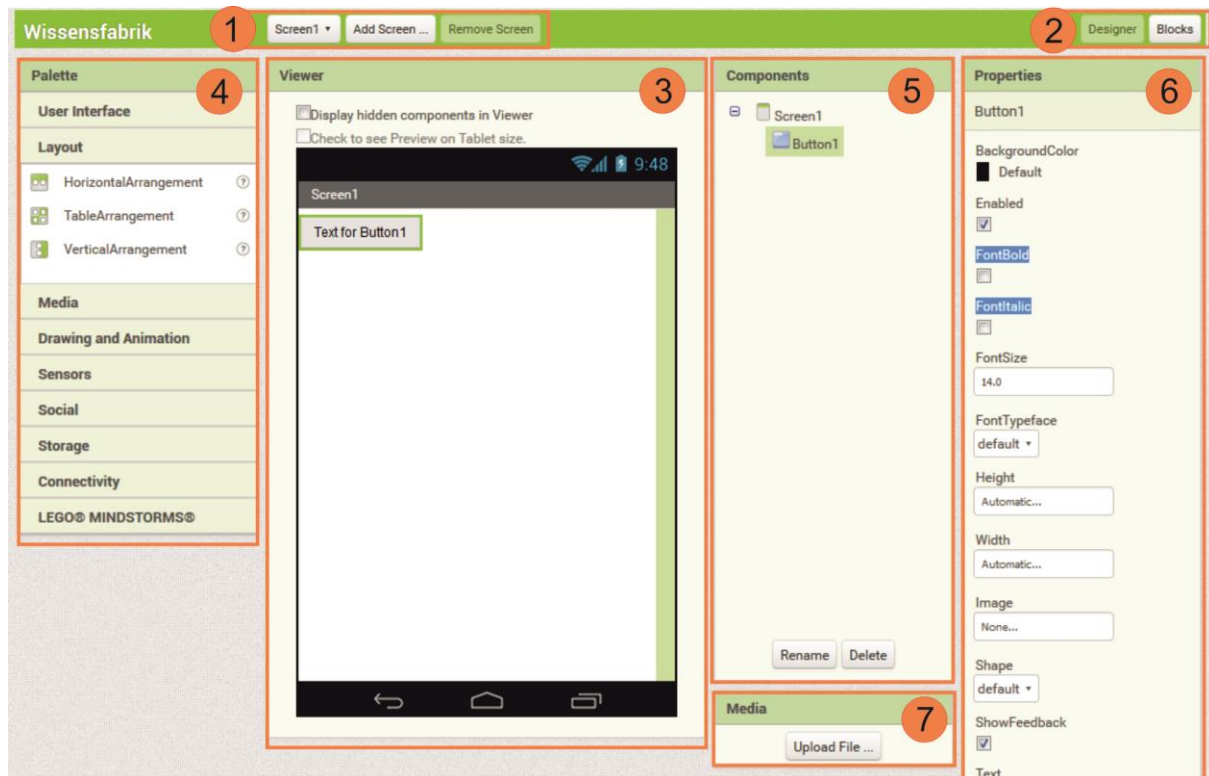
Die Webseite des App Inventors kann unter <http://appinventor.mit.edu> aufgerufen werden. Nach einem Klick auf **Create!** (siehe oben rechts auf der Webseite), wird man zum Login mit einem Google-Account aufgefordert. Nach dem erfolgreichen Login ist eine Übersicht der Projekte zu sehen. Nach Anlegen eines neuen Projekts oder Anklicken eines bestehenden Projekts öffnet sich die Entwicklungsumgebung des App Inventors.

Bevor jedoch mit der eigentlichen App-Entwicklung begonnen werden kann, muss für das Testen der Apps noch ein Emulator-Programm oder eine App zur „Live-Vorschau“ installiert werden. Diese kann auf der Seite <http://appinventor.mit.edu/explore/ai2/setup.html> heruntergeladen werden.

### 5.2 Die grafische Benutzeroberfläche

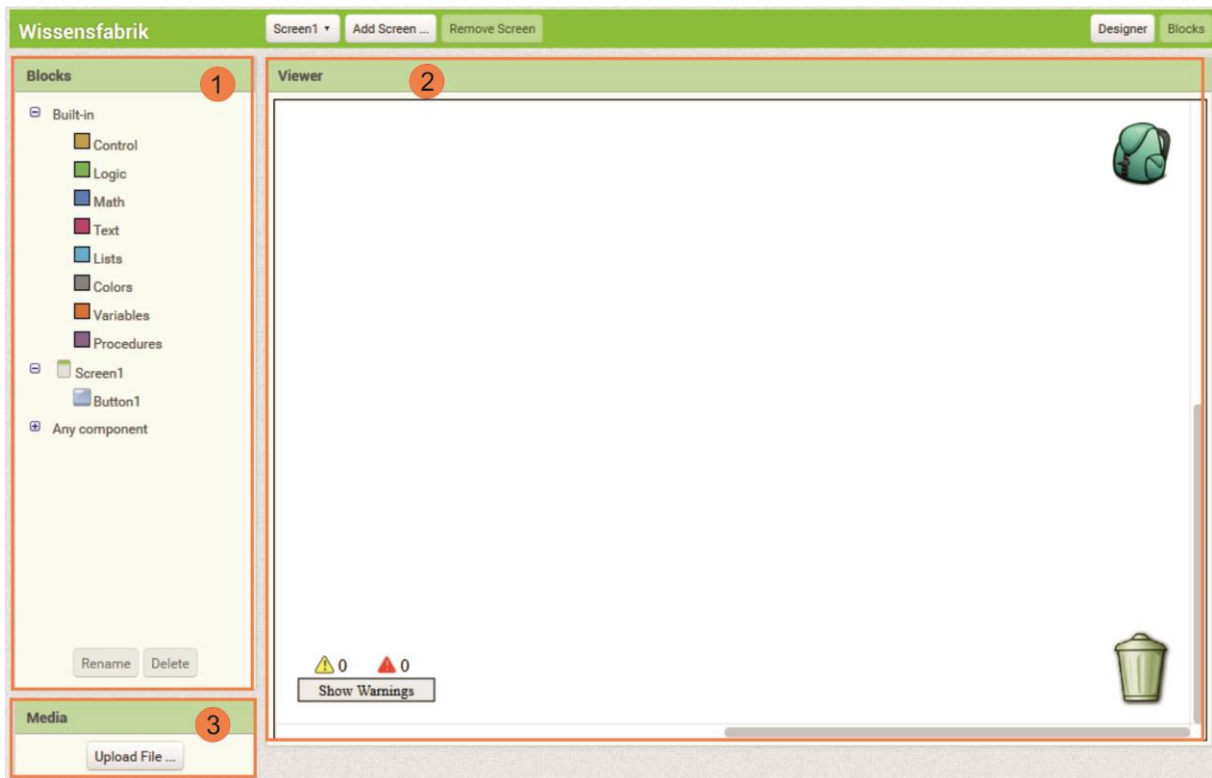
Der App Inventor verfügt über zwei Editoren. Im *Design Editor* werden die einzelnen App-Komponenten (Buttons, Texte, Bilder, Kamera, verschiedene Sensoren etc.) hinzugefügt, gelöscht, bearbeitet und angeordnet. Im *Blocks Editor* erfolgt die Programmierung der zuvor eingefügten Komponenten.

## 5.2.1 Entwicklungsumgebung Design Editor



1. In diesem Bereich sind allgemeine Optionen zum Projekt zu finden. Es kann gespeichert oder für die Installation auf dem Smartphone exportiert werden. Unter *Connect* sind die verschiedenen Möglichkeiten aufgelistet, um die App zu testen. Sie kann entweder mithilfe eines eigenen Programms auf dem Computer emuliert werden oder auf dem Smartphone über die App in einer Vorschau-Version aufgerufen werden.
2. Hier kann zwischen den beiden Editoren gewechselt werden.
3. Der *Viewer*-Bereich zeigt, wie die App grafisch aufgebaut ist. Sichtbare Komponenten wie Buttons, Texte oder Bilder sind hier dargestellt. Somit ist annähernd das zu sehen, was auch die Nutzer der fertigen App auf dem Smartphone angezeigt bekommen.
4. Innerhalb der in diesem Bereich aufgelisteten Kategorien befinden sich die einzelnen Komponenten, die zu der App hinzugefügt werden können, unter anderem:
  - *User Interface* enthält Komponenten, die zur Ein- und Ausgabe sowie zur Interaktion mit dem Nutzer dienen. Enthalten sind beispielsweise *Button*, *CheckBox*, *DatePicker* (Auswahlmöglichkeit für das Datum), *ListPicker* (Auswahllisten) etc.
  - *Sensors* enthält Komponenten, die Sensoren des Smartphones darstellen. Enthalten sind beispielsweise *AccelerometerSensor* (Geschwindigkeitssensor), *BarcodeScanner*, *LocationsSensor* (Positionssensor) etc.
5. Im Bereich *Components* sind alle sichtbaren und nicht sichtbaren Komponenten der App aufgelistet. Hier können auch Bezeichnungen geändert oder Komponenten gelöscht werden.
6. Sobald in Bereich *Viewer* oder *Components* eine Komponente ausgewählt wurde, können unter *Properties* Eigenschaften dieser Komponente verändert werden. So lässt sich beispielsweise die Höhe oder Breite sowie die Schriftfarbe eines Buttons oder Labels bestimmen.
7. Hier können eigene Medien wie Bilder und Musik hochgeladen werden, um sie dann in der App (als Hintergrundbild oder für die Soundausgabe) zu nutzen.

## 5.2.2 Entwicklungsumgebung Blocks Editor



1. Der Bereich *Blocks* beinhaltet Bausteine zur Programmierung der App. Sie sind aufrufbar, indem man mit dem Mauszeiger über eine Kategorie (z. B. *Control*, *Logic*) oder Komponenten (z. B. *Screen1*) fährt.
2. In den *Viewer*-Bereich müssen die einzelnen Bausteine per Drag-and-Drop gezogen und hier verbunden werden. Ein Rechtsklick ermöglicht das schnelle Duplizieren von Bausteinen (was ein erneutes Hineinziehen deutlich verkürzt).
3. In *Media* können eigene Medien hochgeladen werden, wie es bereits aus dem *Design Editor* bekannt.

## 6 Unterrichtliche Umsetzung

Für den Einstieg eignet sich eine Präsentation fertiger Projekte, wobei meist das Smartphone selbst schon eine Motivation für die Schülerinnen und Schüler darstellt. Für die Präsentation können Projekte anderer Schülergruppen oder auch unsere Beispiele genutzt werden. Im Anschluss können die Schülerinnen und Schüler schrittweise mit der Entwicklungsumgebung des App Inventors und den einzelnen Bausteinen vertraut gemacht werden. Im weiteren Verlauf können sie eigene Ideen für Projekte entwickeln und umsetzen.

Die Grundlagen für dieses Modul und die von den Schülerinnen und Schülern zu entwickelnde App stammen aus einem Modul<sup>1</sup> des Schülerlabors der RWTH Aachen.

<sup>1</sup> siehe <http://schuelerlabor.informatik.rwth-aachen.de/modul/einstieg-den-app-inventor>



## 6.1 Grober Unterrichtsplan

Unterrichtsszenarien	Kurze Zusammenfassung
Einführung, Kennenlernen und Entwicklung einer App	Anhand eines Quiz-Tutorials und/oder Lernkarten lernen die Schülerinnen und Schüler die Entwicklungsumgebung des App Inventors kennen und entwickeln Schritt für Schritt ihre erste App. Die Schülerinnen und Schüler erwerben so Grundkenntnisse in der Entwicklung von Applikationen mithilfe des App Inventors.
Vertiefung: Eigene Projekte	Eigene Ideen und Projekte werden von den Schülerinnen und Schülern entwickelt und anschließend mit dem App Inventor umgesetzt.

## 6.2 Stundenverlaufsskizzen

### 6.2.1 Einführung, Kennenlernen und Entwicklung einer App

Die Umsetzung dieses Unterrichtsszenarios benötigt keine feste Vorgabe. Daher soll hier lediglich auf die Handreichung für die Schülerinnen und Schüler (B7.1) sowie das Quiz-Tutorial (B7.2) und die Lernkarten (B7.3) zur Entwicklung einer ersten größeren App verwiesen werden. Die Schülerinnen und Schüler können mithilfe des Quiz-Tutorials eine erste, ganz einfache App selbst umsetzen. Mit Hilfe der Lernkarten gestalten sie eine App, mit der sich auf Knopfdruck ein Foto aufnehmen lässt, das anschließend durch Wischbewegungen mit dem Finger bemalt werden kann. Die Schülerinnen und Schüler benötigen dafür etwa zwei bis drei Einzelstunden. Vor dem Einsatz der Lernkarten sollte die Lehrkraft selbst einen kurzen Überblick zu der Entwicklungsumgebung geben sowie den Wechsel zwischen und Zusammenhang von Design und Blocks Editor erläutern.

### 6.2.2 Vertiefung: Eigene Projekte

Um die Kreativität der Schülerinnen und Schüler zu fördern und den App Inventor auch als Werkzeug für den eigenen Fachunterricht zu nutzen, können Sie die Schülerinnen und Schüler eigene Apps entwerfen lassen. Dabei können natürlich auch Vorgaben erteilt werden, etwa dass sich die zu entwickelnde App auf den Fachunterricht beziehen muss.

## 7 Einbettung in verschiedene Fächer und Themen

Da in diesem Modul die Entwicklung und Programmierung einer App für das Smartphone im Vordergrund steht, beziehen sich die zuzuordnenden Kompetenzen hauptsächlich auf die Informatik.

Jedoch bedeutet dies nicht, dass das Modul ausschließlich für den Informatik- oder Technikunterricht geeignet ist. Schon nach kurzer Zeit und nach Abschluss des Tutorials können die erworbenen Kenntnisse genutzt werden, um kleinere Anwendungen für Fachunterricht jenseits der Informatik und Technik zu realisieren. Beispielsweise lassen sich mit dem Wissen aus den Tutorial Werkzeugen für den Mathematikunterricht oder Vokabeltrainer für den Fremdsprachenunterricht erstellen. Zwar werden so kaum neue fachbezogene Kompetenzen vermittelt, aber die Planung und Entwicklung der App festigen bestehende Kompetenzen, weil die Schülerinnen und Schüler erworbenes Wissen reflektieren.



Die folgenden Kompetenzen finden sich entweder in den Bildungsstandards der Kultusministerkonferenz oder in den einzelnen Rahmenlehrplänen der Länder wieder:

### Informatik

Die Schülerinnen und Schüler ...

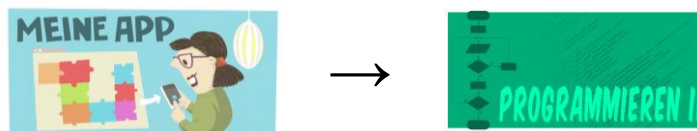
- erstellen Produkte unter Anwendung fortgeschrittener Techniken von Standardsoftware; falls es das Produkt erfordert, arbeiten sie sich in geringem Umfang in Spezialsoftware ein.
- erwerben beim Bearbeiten von Softwareprojekten in angemessenem Umfang Kenntnisse über Analyse- und Modellierungsverfahren sowie Projektmanagement.
- verstehen Programmabläufe und die Arbeitsweise von Schnittstellen.
- können erdachte Systeme in technische übertragen.
- kennen sich in Programmierumgebungen aus.
- gestalten eine Spielumgebung entsprechend der Spielidee.
- entwerfen und implementieren eine Spielvariante mit informatischen Werkzeugen.

## 8 Anschluss Themen

Als Anschluss Themen im Zusammenhang mit IT2School bieten sich folgende Module an:

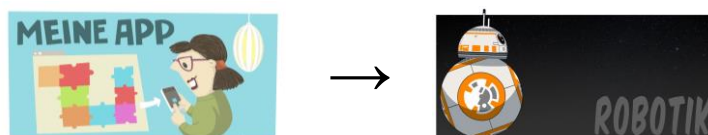
### Beispiel: Programmierung

Das Aufbaumodul A3 bietet eine gute Möglichkeit zur Vertiefung der Programmierkenntnisse. In diesem Modul lernen die Schülerinnen und Schüler die textuelle Programmiersprache Python kennen und vertiefen ihr Geschick in der Programmierung.



### Beispiel: It spielend entdecken

Möchten Sie ein nächstes kreatives Projekt mit ihren Schülerinnen und Schülern durchführen, empfehlen wir Ihnen das Modul A4 *Robotik*. Die Schülerinnen und Schüler bauen dabei aus Alltagsgegenständen einen Roboter und programmieren ihn mit dem App Inventor oder Arduino.



## 9 Literatur und Links

- Appinventor.org. Unterrichtsmaterialien und Kurse zum App Inventor. Online: <http://www.appinventor.org/content/CourseInABox/web/CCStockMarket>
- Bopp, Julian (2015): Apps entwickeln wie DU willst – mit dem MIT App Inventor 2. Ideen für den Unterricht. Online:




[https://www.juforum.de/fileadmin/downloads/oeffentlich/2014/App\\_Inventor\\_Tutorial\\_2.pdf](https://www.juforum.de/fileadmin/downloads/oeffentlich/2014/App_Inventor_Tutorial_2.pdf)

- RWTH Aachen: <http://schuelerlabor.informatik.rwth-aachen.de/modul/einstieg-den-app-inventor>

## 10 Arbeitsmaterialien

Nr.	Titel	Beschreibung
 B7.1	Einstieg in den App Inventor	Handreichung für den Einstieg, die in Kürze die Anmeldung beim App Inventor, den Aufbau sowie das Testen der Programme erläutert.
 B7.2	Quiz-Tutorial	Tutorial für eine einfache Quiz-App
 B7.3	Lernkarten	Lernkarten zur selbstgesteuerten Vertiefung.
 B7.4 L	BlueCoLight	Erläuterung zum BlueCoLight sowie einige Beispielprojekte zum umsetzen.
 B7.4 SuS	BlueCoLight	Erläuterungen zum BlueCoLight und zum App Inventor Projekt zum erstellen einer eigenen App für den BlueCoLight.

### Legende

-  Material für Schülerinnen und Schüler
-  Material für Lehrkräfte sowie Unternehmensvertreterinnen und Unternehmensvertreter
-  Zusatzmaterial

## 11 Glossar

Begriff	Erläuterung
Android	Bezeichnet ein Betriebssystem für Smartphone- und Tablet-Systeme von Google. Es handelt sich dabei um eine freie und quelloffene Software, die auf der Grundlage von Linux entwickelt wurde.
App	Kurzbezeichnung für Applikation.
Bluetooth	Bluetooth wurde in den 1990er Jahren durch die Bluetooth Special Interest Group (SIG) entwickelt und ist seitdem ein Industriestandard gemäß IEEE 802.15.1 für die Übertragung von Daten zwischen Geräten über kurze Distanz per Funktechnik (WPAN). Der Name „Bluetooth“ soll an den dänischen König



	Harald Blauzahn (englisch Harald Bluetooth) erinnern, der verfeindete Teile von Norwegen und Dänemark vereinte.
Drag-and-Drop	Oft auch Drag & Drop oder Drag'n'Drop, zu deutsch: „Ziehen und Ablegen“, stellt eine Methode zum Bewegen von Elementen auf grafischen Benutzeroberflächen dar. Ein Element kann dabei mithilfe eines Zeigegerätes (z. B. Maus) gezogen und an seinem Bestimmungsort losgelassen werden. Allgemeinen wird Drag-and-Drop genutzt, um Aktionen auszuführen oder Beziehungen zwischen Objekten herzustellen.
Editor	Engl. <i>to edit</i> . (Texte etc.) herausgeben, redigieren, bearbeiten, (Filme) schneiden. Ein Editor ist ein Computerprogramm, das zur Erstellung und Bearbeitung von digitalen Daten (Dateien) verwendet wird (deutsch auch: Bearbeitungsprogramm).
Emulator	Software-Emulatoren sind Programme, die Eigenschaften eines Computers oder eines Handys nachbilden und es so ermöglichen, Software für die Zielhardware auf einem Computer mit einer anderen Architektur zu verwenden und zu testen. So können z. B. Spiele für ältere Spielekonsolen auch auf einem PC oder einer neueren Spielekonsole gespielt werden oder Handyapps auf dem Computer getestet werden, ohne jedes Mal ein Handy benutzen zu müssen.
Google-Account	Auch Google-Konto; ein bei Google registriertes Benutzerkonto, mit dem verschiedene Google-Dienste genutzt werden können. In dem Konto werden z. B. personenbezogene Daten und Informationen gespeichert und verknüpft. Mit einem Google-Konto können Nutzer bei verschiedenen Anwendungen, auch bei Drittanbietern, anmelden, ohne jedes Mal neue Zugangsdaten eingeben zu müssen.