

(Informatische) Wissensvermittlung in der Grundschule

Workshop für Coaches des GTA „Programmieren mit Calliope mini“



Materialien zum Download



proGTA.de



Das Projekt "Programmieren mit Calliope mini" wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Wie geht ihr an die Planung heran?

Welche Herausforderungen habt ihr aktuell?

Welche Fragen zur Planung/Durchführung habt ihr mitgebracht?



proGTA.de



Das Projekt "Programmieren mit Calliope mini" wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Themenschwerpunkte

- Planung von Lerneinheiten
- Methoden der direkten Instruktion und Projektunterricht
- Informatikkonzepte verstehen



Planung von Lerneinheiten

- Kindgerecht und entwicklungsförderlich
- Orientiert an Interessen, Möglichkeiten und Bedürfnissen der Kinder
- Herausforderung:
 - Kinder sollen selbstständig und selbsttätig arbeiten, nicht unter- nicht überfordern
 - Heterogene Gruppen: Unterschiedliche Bedürfnisse nach Instruktion/Begleitung, Motivationen zur Teilnahme

→ Geheimrezept für Lerneinheiten gibt es nicht, lediglich Grundsätze und Hilfen, mit denen man sich zum Ideal annähern kann!



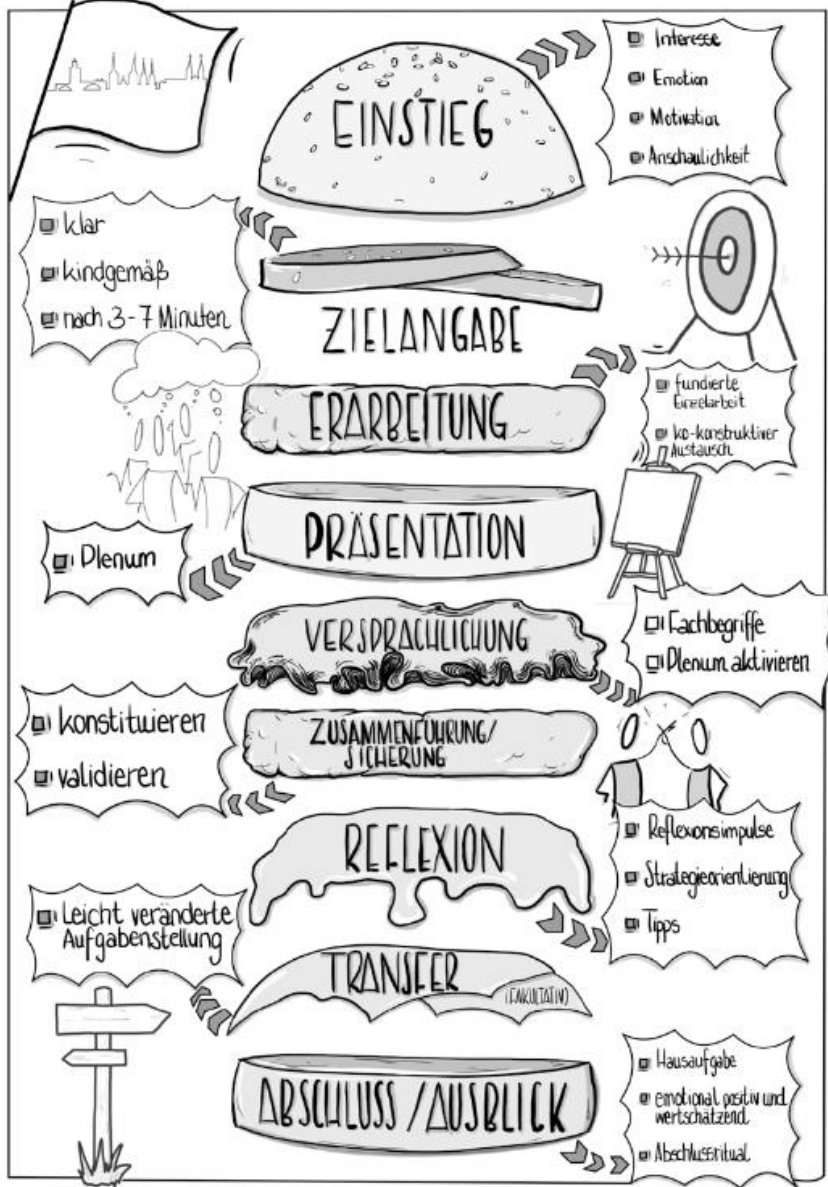


Abbildung 2: Der »Bamburger« der Unterrichtsplanung

Köhler/Weiß: S. 13

Ein Deckblatt zur täglichen Unterrichtsplanung

3



Tägliche Unterrichtsvorbereitung	Klasse	Fach	Lehrplanzuordnung	Datum	Zeit
Titel der Sequenz					
Stundenthema					Nr.
Zielsetzung der Unterrichtseinheit					
Das muss vorher geschehen sein:					
Hausaufgabe:					
Meine Arbeitsmittel:			Arbeitsmittel der Schüler:		
Meine Reflexionsimpulse für die Schüler:					
Reflexion: Die drei besten Aspekte der Unterrichtseinheit			Darauf achte ich das nächste Mal besonders:		
1.					
2.					
3.					

Köhler/Weiß: S. 15

Alles Notwendige der Planung für eine gelungene Lerneinheit

Zielangabe der Stunde nach ca. 3-7 Minuten!

Bei der Planung unbedingt Puffer lassen, denn es geht mehr Zeit zum Hochherunterfahren des Computers, Auspacken Calliope etc. drauf, als man denkt



Direkte Instruktion

- Lehrende bereitet den Lernprozess vor und begleitet ihn durch Vormachen, Erklären und Strukturieren des Lernstoffes
- Sachverhalte werden kleinschrittig und systematisch angeleitet
- Aufteilung in 7 Phasen, Phasenlänge kann variieren

- Nachteile: starker Fokus auf Lehrperson, Selbständigkeit der Schüler:innen wird eingeschränkt



1.	Vorbereitung: Die Lehrkraft analysiert den Lerninhalt und teilt ihn in sehr kleine einzelne Lernschritte auf.	Programmierungsumgebung Kennenlernen (MakeCode)
2.	Einführung: Das Thema und das Ziel der Einheit werden von der Lehrkraft explizit verdeutlicht.	Ziel: Bedienung MakeCode
3.	Entwicklung: Die Lehrkraft demonstriert den Schüler*innen die einzelnen Lernschritte (die Modellierung findet detailliert, strukturiert und mit positiven, eindeutigen Beispielen statt [...])	Computer anschalten, Anmelden, Browser öffnen, Internetadresse eingeben, neues Projekt, leere Projektseite
4.	Angeleitete Praxis: Alle Schüler*innen führen nach und nach genau die Lernschritte aus, die die Lehrkraft einzeln vormacht [...]; die Lehrkraft überprüft, ob alle die Schritte nachvollziehen können und gibt ggf. gezielt Hilfestellung.	Wenn alle bei der leeren Projektseite angekommen sind: „Was könnt ihr erkennen?“ (Werkzeuge, Simulator, Stop, Lautlos...)
5.	Zusammenfassung: Die Lehrkraft [(oder Schüler:in)] fasst abschließend den gelernten Inhalt zusammen und hebt das Ziel hervor.	Sicherung z.B. mit Arbeitsblatt oder im Arbeitsheft
6.	Unabhängige Praxis: Die Schüler*innen üben und vertiefen mit verschiedenen Methoden und anhand verschiedener Schwierigkeitsniveaus das Gelernte.	Kleinen Code programmieren lassen, z.B. LED-Matrix, am Simulator die Veränderungen beobachten lassen
7.	Evaluation: Haben alle Schüler*innen von der Lernphase profitiert? In der Phase spielt wirksames Feedback eine wichtige Rolle [...].	



Projektunterricht

- Eine Form des offenen Unterrichts
- Schüler:innen haben mehr Lernautonomie, Lernen wird selbst in die Hand genommen
- Charakteristische Merkmale des offenen Unterrichts:
 - anregende Lernumwelt
 - Lernorganisation ist frei und flexibel mit wenigen frontalen Phasen
 - Methoden sind kreativ, selbsttätig und entdeckend
 - vertrauensvolle Atmosphäre
 - Dokumentation der Ergebnisse
- Nachteil: gewisse Voraussetzungen (Lernstrategien) erforderlich, Schwierig für Kinder mit Lern- und Konzentrationsschwächen, Forschung zur Wirksamkeit nicht eindeutig



1.	Projektinitiative ausgehend von einem Mitglied der Projektgruppe	Bewässerungsanlage für Schulpflanzen
2.	Auseinandersetzung mit der Initiative mit dem Ergebnis einer Projektskizze	Pflanzen sollen Signale geben wenn die Erde zu trocken wird; Schleuse aus dem Bewässerungstank wird geöffnet und geschlossen
3.	Gemeinsame Entwicklung des Betätigungsfelds mit der Erstellung eines Projektplans	Wie muss es programmiert werden? Welche Bauteile braucht man? Wie wird das zusammengebaut?
4.	Aktivitäten: Durchführung des Projekts	Prototyp bauen (z.B. an einer Pflanze)
5.	Beendigung durch Bewussten Abschluss, Rückkopplung auf Projektinitiative oder Auslaufenlassen	Präsentation beim Tag der offenen Tür, Wandplakat erstellen
6.	Fixpunkte (bereits Geschehenes wird geprüft und angepasst)	Zeitlich unabhängig, während aller Prozesse
7.	Metainteraktion (Konflikte und deren Klärung)	



Informatikkonzepte verstehen

Grundsätze für das Lehren von Computer Science

1. Ditch Uniformity

- SuS lernen in unterschiedlichen Tempi
- Alle gleichzeitig zum Ziel zu bringen --> einfach nicht realistisch und kann sowohl die langsamen abschrecken und die schnellen langweilen
- Beim GTA: Etappenziele setzen, jede/r kommt da zu unterschiedlichen Zeiten hin

2. Change of view

- Eine Zwischenaktivität machen, die nicht Programmieren ist, tut gut!
- Kleine Pausen machen, nicht 90 min am Stück an den Bildschirm "fesseln" -> 20/20/20 Regel (nach 20' min, 20 Meter weit schauen und für 20 Sekunden)
- Bewegungspausen/Strecken gehören zum Prozess dazu

3. Collaborate

- Zusammenarbeit oder schauen, wie andere das Problem gelöst haben, gehört zum realen Arbeitsprozess im Programmierumfeld
- Kollaboratives Arbeiten schult zudem Gruppenarbeitskompetenz und schweißt die GTA Gruppe zusammen

4. Don't be a Know-It-All

- Lehrer heißt nicht (nur) Experte, eher Unterstützer und Begleiter
- Wenn SuS Projekte anstreben, die über das eigene Verständnis hinausgehen --> ermutigen, und sich von S erklären lassen

Nicht zu schnell nach Hilfe bitten: Try Three, Try Three, Ask me -> 3x alleine versuchen und scheitern, 3x mit einem anderen S versuchen und scheitern, dann erst Hilfe bei Lehrkraft fragen

Prottzman: S. 60-63.

Möglicher Lehrweg	Aufmerksamkeitsrichtungen	Beispiel: Vom Pixel zum Binärcode (vgl. Köhler/Schmid/Weiß/Weitz 2020)
Anknüpfen an kindliche Vorerfahrungen	<ul style="list-style-type: none"> Kindgerechte Beispiele aus dem Alltag der Kinder fokussieren die Kinder auf das Ziel der Unterrichtsstunde 	<ul style="list-style-type: none"> Unterschiede zwischen digitalen und analogen Bildern feststellen Erste Erfahrungen von Unterschieden »digital« und »analog« machen
Zielangabe	<ul style="list-style-type: none"> Klare Formulierung in kindgerechter Form 	<ul style="list-style-type: none"> Impuls: »Was sind Pixel?«
Direkte Instruktion	<ul style="list-style-type: none"> Die Sachinformation lässt sich am ökonomischsten in direkter Instruktion erschließen Wichtig ist die Erklärung in der Regel durch die Lehrkraft, ggf. auch in für Grundschüler didaktisch reduzierter textgestützter Form und mit Abbildungen angereicherter Form 	<ul style="list-style-type: none"> Als Pixel bezeichnet man den kleinsten Teil eines digitalen Bildes Pixel ist das englische Wort für »Bildpunkt« oder »Bildelement« Ein Pixel wird meistens als Quadrat dargestellt. Die Seitenlänge des Quadrats kann ganz unterschiedlich sein Auf dem Computerbildschirm wird alles, egal ob es ein Text ist oder ein Bild, mit Pixeln, also Bildpunkten, dargestellt Wenn viele Bildpunkte ganz dicht nebeneinander und untereinander stehen, macht unser Gehirn daraus ein Gesamtbild
Anwendung Analoge Handlungsorientierung Gegebenenfalls digitale Handlungsorientierung	<ul style="list-style-type: none"> Das »analoge« Lernen ist die passende Vorstufe, die das Verstehen informatorischer Grundkonzepte zu verstehen, unterstützt Erst nach analoger Handlungsorientierung erfolgt der gezielte Einsatz digitaler Medien 	<ul style="list-style-type: none"> Pixel-Rätsel Schwarz-weiße Bilder im Pixelraster Pixelbilder erraten und erstellen Großflächige Pixelbilder mit Haftnotizblockzetteln gestalten Herausarbeiten des Zusammenhangs zwischen Bildschärfe und Auflösung Digitale Malprogramme nutzen, Bilder gestalten Analogbilder digitalisieren
Konzepttransparenz	<ul style="list-style-type: none"> Rückbezug des Unterrichtsschwerpunktes auf das informatorische Konzept nehmen Versprachlichen der Gemeinsamkeiten und der Unterschiede zwischen den analogen und digitalen Beispielen aus der Handlungsorientierung der Anwendungsphase 	<ul style="list-style-type: none"> Mit digitalen Malprogrammen gestaltete Bilder zoomen und Pixel sichtbar machen

Köhler/Weiß: S. 195



proGTA.de



Das Projekt "Programmieren mit Calliope mini" wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Zusammenfassung

- Inhalt in kleine Schritte aufteilen → weniger ist mehr!
- Erweiterungsaufgaben für „Schnelle“
- Klarheit schaffen: Zielstellungen konkretisieren

Was?	»Je nach individuellem Leistungsvermögen« [prozessbezogener Inhalt und Anspruchsniveau]
Wie/Womit?	»indem sie« [Methoden-Operator]
Wozu?	»um ... zu« [Begründung: in der Regel abgeleitet und konkretisiert aus den Kompetenzerwartungen des Lehrplans]

Köhler/Weiß: S. 214



Literaturverzeichnis

- Köhler/Weiß: Die tägliche Unterrichtsvorbereitung für die Grundschule. Beltz-Verlag, Weinheim Basel, 2022.
- Bartels/Vierbuchen: Einführung in die Grundschulpädagogik. Kohlhammer, Stuttgart, 2022.
- Seifert/Wiedenhorn: Grundschulpädagogik. Verlag Ferdinand Schöningh, Paderborn, 2018.
- Prottzman: Computer science for the elementary classroom. In: acm Inroads (volume 5/ issue 4), New York, 2014, S. 60-63.

