

Mobile Informatiksysteme im Unterricht

Daniel Spittank

Donnerstag, 26. Juni 2014

Inhalt

1 Einleitung

- Inhalt
- Ausgangslage
- Motivation
- Alltagsbezug
- Gesellschaftliche Reaktionen

2 Einsatzmöglichkeiten

- Informatikunterricht
- Weitere Nutzungsmöglichkeiten in der Schule

3 Kriterien

- Zugänge

4 Android

- Scripting Layer for Android
- QPython
- API

5 Umsetzung

- API-Wrapper
- Erfahrungen
- Online-Material

Informatisierung der Gesellschaft

- Informatiksysteme werden immer weiter entwickelt.
- Informatik durchdringt den Alltag.
- Die moderne Welt ist ohne Informatik kaum vorstellbar.

Informatisierung der Gesellschaft

- Informatiksysteme werden immer weiter entwickelt.
- Informatik durchdringt den Alltag.
- Die moderne Welt ist ohne Informatik kaum vorstellbar.
- Informatiksysteme werden mobil.
- Mobile Informatiksysteme sind überall anzutreffen.

Informatisierung der Gesellschaft

- Informatiksysteme werden immer weiter entwickelt.
- Informatik durchdringt den Alltag.
- Die moderne Welt ist ohne Informatik kaum vorstellbar.
- Informatiksysteme werden mobil.
- Mobile Informatiksysteme sind überall anzutreffen.
- Klassische, stationäre Informatiksysteme verlieren an Bedeutung.

Fehlvorstellungen

- **Trotzdem wird Informatik mit Computertechnik gleichgesetzt.**
- Dieses Bild haben insbesondere auch Schüler.
- Schüler werden noch häufig mit Computerkursen konfrontiert.
- Der Informatikunterricht findet im Computerraum statt.
- **Mobile Geräte als Ausweg?**

Vorteile und Hoffnungen

- Stärkerer Alltagsbezug
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

Nachteile und Befürchtungen

- Erschwerte Bedienung
- Verringerte Geschwindigkeit
- ⇒ Frustration?
- Exklusive Nutzung der Geräte

Gerätebesitz

Gerätebesitz nach JIM-Studie 2013 MPFS, 2013

- 99% aller 12- bis 19-jährigen Schüler besitzen ein Mobiltelefon.

Gerätebesitz

Gerätebesitz nach JIM-Studie 2013 MPFS, 2013

- 99% aller 12- bis 19-jährigen Schüler besitzen ein Mobiltelefon.
- Ein stationäres Informatiksystem besitzen hingegen nur 80%.

Gerätebesitz

Gerätebesitz nach JIM-Studie 2013 MPFS, 2013

- 99% aller 12- bis 19-jährigen Schüler besitzen ein Mobiltelefon.
- Ein stationäres Informatiksystem besitzen hingegen nur 80%.
- Außerdem sind Mobiltelefone gleichermaßen bei allen Altersgruppen zu finden.
- Smartphones weniger weit verbreitet (etwa 72%)

Gerätebesitz

Gerätebesitz nach JIM-Studie 2013 MPFS, 2013

- 99% aller 12- bis 19-jährigen Schüler besitzen ein Mobiltelefon.
- Ein stationäres Informatiksystem besitzen hingegen nur 80%.
- Außerdem sind Mobiltelefone gleichermaßen bei allen Altersgruppen zu finden.
- Smartphones weniger weit verbreitet (etwa 72%)
- **Aber die am schnellsten wachsende Gerätegruppe (2011: 20%)**

Verfügbarkeit von Geräten

- Persönliche Geräte der Schülerinnen und Schüler sind vorhanden.
- Telefone der Schülerinnen und Schüler werden immer „smarter“.
- Hersteller drängen mit ihren Angeboten in die Schulen (allen voran Apple mit seinen iPads).
- Schulbuchverlage bieten einzelne Apps und arbeiten an eigener Plattform.

Verfügbarkeit von Geräten

- Persönliche Geräte der Schülerinnen und Schüler sind vorhanden.
- Telefone der Schülerinnen und Schüler werden immer „smarter“.
- Hersteller drängen mit ihren Angeboten in die Schulen (allen voran Apple mit seinen iPads).
- Schulbuchverlage bieten einzelne Apps und arbeiten an eigener Plattform.
- Mobile Informatiksysteme werden also mittelfristig den Weg in die Schulen finden oder sind bereits da.

Verfügbarkeit von Geräten

- Persönliche Geräte der Schülerinnen und Schüler sind vorhanden.
- Telefone der Schülerinnen und Schüler werden immer „smarter“.
- Hersteller drängen mit ihren Angeboten in die Schulen (allen voran Apple mit seinen iPads).
- Schulbuchverlage bieten einzelne Apps und arbeiten an eigener Plattform.
- Mobile Informatiksysteme werden also mittelfristig den Weg in die Schulen finden oder sind bereits da.
- Diese Einführung sollte didaktisch sinnvoll gestaltet werden!

Sorgen

- Mobiltelefone werden häufig mit der Verbreitung von Pornographie und Gewaltvideos in Verbindung gebracht.
- Mobiltelefone werden in erster Linie als Ablenkung vom Lernen betrachtet.

Reaktionen

- Verbote sind an Schulen üblich.
- Teilweise sogar gesetzliche Handyverbote an Schulen (in Bayern seit 2006)

Reaktionen

- Verbote sind an Schulen üblich.
- Teilweise sogar gesetzliche Handyverbote an Schulen (in Bayern seit 2006)
- These: Dies ist dem Ziel der Entwicklung zum mündigen Gesellschaftsmitglied abträglich.

Reaktionen

- Verbote sind an Schulen üblich.
- Teilweise sogar gesetzliche Handyverbote an Schulen (in Bayern seit 2006)
- These: Dies ist dem Ziel der Entwicklung zum mündigen Gesellschaftsmitglied abträglich.
- Besser: Sinnvoll in den Unterricht integrieren und verantwortungsvollen Umgang vermitteln.

Einsatz im Informatikunterricht

- Mobilgeräte sind vollständige Informatiksysteme.
- Lehrpläne sind damit grundsätzlich umsetzbar (vgl. Heming, 2009).

Einsatz im Informatikunterricht

- Mobilgeräte sind vollständige Informatiksysteme.
- Lehrpläne sind damit grundsätzlich umsetzbar (vgl. Heming, 2009).
- Vorteile wie zuvor beschrieben

Einsatz im Informatikunterricht

- Mobilgeräte sind vollständige Informatiksysteme.
- Lehrpläne sind damit grundsätzlich umsetzbar (vgl. Heming, 2009).
- Vorteile wie zuvor beschrieben
- Einige Zugänge eröffnen sich mit Mobilgeräten leichter, z.B.:
 - Datenschutz
 - Kommunikation(-protokolle)
 - Nutzung von Audio (TTS, SR, Mikrofon) und Video (Kamera)
 - Objektkarten ↔ QR-Codes

Einsatz im Informatikunterricht

- Mobilgeräte sind vollständige Informatiksysteme.
- Lehrpläne sind damit grundsätzlich umsetzbar (vgl. Heming, 2009).
- Vorteile wie zuvor beschrieben
- Einige Zugänge eröffnen sich mit Mobilgeräten leichter, z.B.:
 - Datenschutz
 - Kommunikation(-protokolle)
 - Nutzung von Audio (TTS, SR, Mikrofon) und Video (Kamera)
 - Objektkarten ↔ QR-Codes
- Außerdem etwa: Modellierung und Simulation von Automaten

Unterrichtseinsatz

- Allgemein
 - Recherche
 - Audiovisuelle Dokumentation
 - Quiz
 - Wissensnetze
 - Notizen
 - (Präsentation)

Unterrichtseinsatz

- Allgemein
 - Recherche
 - Audiovisuelle Dokumentation
 - Quiz
 - Wissensnetze
 - Notizen
 - (Präsentation)
- Sozialwissenschaften
 - Interviews und Umfragen
 - Statistiken
 - Umgang mit Medien (als Informationsquellen)

Unterrichtseinsatz

- Allgemein
 - Recherche
 - Audiovisuelle Dokumentation
 - Quiz
 - Wissensnetze
 - Notizen
 - (Präsentation)
- Sozialwissenschaften
 - Interviews und Umfragen
 - Statistiken
 - Umgang mit Medien (als Informationsquellen)
- Mathematik und NW
 - Ersatz für GTR
 - Mathematikapps
 - Simulationen
 - GPS

Unterrichtseinsatz

- Allgemein
 - Recherche
 - Audiovisuelle Dokumentation
 - Quiz
 - Wissensnetze
 - Notizen
 - (Präsentation)
- Sozialwissenschaften
 - Interviews und Umfragen
 - Statistiken
 - Umgang mit Medien (als Informationsquellen)
- Mathematik und NW
 - Ersatz für GTR
 - Mathematikapps
 - Simulationen
 - GPS
- Sprachen
 - SMS-Stil
 - Vokabeln
 - Aussprache

Unterrichtseinsatz

- Allgemein
 - Recherche
 - Audiovisuelle Dokumentation
 - Quiz
 - Wissensnetze
 - Notizen
 - (Präsentation)
- Sozialwissenschaften
 - Interviews und Umfragen
 - Statistiken
 - Umgang mit Medien (als Informationsquellen)
- Mathematik und NW
 - Ersatz für GTR
 - Mathematikapps
 - Simulationen
 - GPS
- Sprachen
 - SMS-Stil
 - Vokabeln
 - Aussprache
- Geographie
 - GPS
 - Kartenmaterial

Verschiedene Perspektiven

- Akzeptanz der Wirklichkeit (vgl. Spittank, 2012)

Verschiedene Perspektiven

- Akzeptanz der Wirklichkeit (vgl. Spittank, 2012)
- Analyse der Wirklichkeit (vgl. Heming, 2009)

Verschiedene Perspektiven

- Akzeptanz der Wirklichkeit (vgl. Spittank, 2012)
- Analyse der Wirklichkeit (vgl. Heming, 2009)
- Veränderung der Wirklichkeit (vgl. Heming, 2009)

Verschiedene Perspektiven

- Akzeptanz der Wirklichkeit (vgl. Spittank, 2012)
- Analyse der Wirklichkeit (vgl. Heming, 2009)
- Veränderung der Wirklichkeit (vgl. Heming, 2009)
 - Entwicklung **für** die Geräte (vgl. Carrie, 2006)

Verschiedene Perspektiven

- Akzeptanz der Wirklichkeit (vgl. Spittank, 2012)
- Analyse der Wirklichkeit (vgl. Heming, 2009)
- Veränderung der Wirklichkeit (vgl. Heming, 2009)
 - Entwicklung **für** die Geräte (vgl. Carrie, 2006)
 - Entwicklung **mit** den Geräten (vgl. Carrie, 2006)
 - Automatisierung
 - Skripte
 - Apps
 - Geschlossene Laufzeitumgebungen
 - Web-Apps

Gerätetypen

Merkmal	Handys	Smartphones	Hybride	Tablets
Progammierbarkeit	⊖	⊕	⊕	⊕
Bedienung	⊖	⊙	⊕	⊕
Alltagsbezug	⊕	⊕	⊙	⊖
Mobilität	⊕	⊕	⊕	⊖
Ausstattung	⊖	⊙	⊕	⊕
Kosten	⊕	⊙	⊖	⊖

⊕ = geeignet, ⊙ = eingeschränkt geeignet, ⊖ = ungeeignet

Auswahlkriterien

Kriterien für die Geräteauswahl

Notwendig

- Sinnvolle Einbindung
- Alltagsbezug und Verbreitung
- Programmierbarkeit
- Verfügbarkeit von Werkzeugen und Dokumentation
- Kosten

■ Langlebigkeit

Auswahlkriterien

Kriterien für die Geräteauswahl

Notwendig

- Sinnvolle Einbindung
- Alltagsbezug und Verbreitung
- Programmierbarkeit
- Verfügbarkeit von Werkzeugen und Dokumentation
- Kosten

- Langlebigkeit

Wünschenswert

- Gute Ausstattung
- Freie Software
- Umweltfreundlichkeit und Sozialverträglichkeit

Auswahlkriterien

Kriterien für die Geräteauswahl

Notwendig

- Sinnvolle Einbindung
- Alltagsbezug und Verbreitung
- Programmierbarkeit
- Verfügbarkeit von Werkzeugen und Dokumentation
- Kosten

- Langlebigkeit

Wünschenswert

- Gute Ausstattung
- Freie Software
- Umweltfreundlichkeit und Sozialverträglichkeit

Fazit: Von den relevanten Plattformen bleibt bisher nur Android übrig.

Vorteile

- SL4A ermöglicht Python und weitere Skriptsprachen,

Vorteile

- SL4A ermöglicht Python und weitere Skriptsprachen,
- ermöglicht sowohl direkte Ausführung als auch Erstellung von Apps.
- Ausführung direkt oder getriggert möglich

Vorteile

- SL4A ermöglicht Python und weitere Skriptsprachen,
- ermöglicht sowohl direkte Ausführung als auch Erstellung von Apps.
- Ausführung direkt oder getriggert möglich
- API verfügbar

Vorteile

- SL4A ermöglicht Python und weitere Skriptsprachen,
- ermöglicht sowohl direkte Ausführung als auch Erstellung von Apps.
- Ausführung direkt oder getriggert möglich
- API verfügbar
- Setuptools

Nachteile

- API ist nicht objektorientiert.
- API ist nicht intuitiv, nicht einheitlich und zu aufwendig.

Nachteile

- API ist nicht objektorientiert.
- API ist nicht intuitiv, nicht einheitlich und zu aufwendig.
- leider teilweise schlecht dokumentiert

Nachteile

- API ist nicht objektorientiert.
- API ist nicht intuitiv, nicht einheitlich und zu aufwendig.
- leider teilweise schlecht dokumentiert
- Interaktiver Python-Shell fehlt Vervollständigung etc.

Nachteile

- API ist nicht objektorientiert.
- API ist nicht intuitiv, nicht einheitlich und zu aufwendig.
- leider teilweise schlecht dokumentiert
- Interaktiver Python-Shell fehlt Vervollständigung etc.
- UI-Chaos

Vorteile

- Integriert SL4A und diverse Python-Bibliotheken,

Vorteile

- Integriert SL4A und diverse Python-Bibliotheken,
- ermöglicht sowohl direkte Ausführung als auch Erstellung von Apps,

Vorteile

- Integriert SL4A und diverse Python-Bibliotheken,
- ermöglicht sowohl direkte Ausführung als auch Erstellung von Apps,
- bietet eine vollständige Entwicklungsumgebung.

Vorteile

- Integriert SL4A und diverse Python-Bibliotheken,
- ermöglicht sowohl direkte Ausführung als auch Erstellung von Apps,
- bietet eine vollständige Entwicklungsumgebung.
- einheitliche UI-Entwicklung über Kivy und pygame möglich

Nachteile

- unklare Lizenzsituation

Nachteile

- unklare Lizenzsituation
- Kommunikation mit den Entwicklern schwierig

Nachteile

- unklare Lizenzsituation
- Kommunikation mit den Entwicklern schwierig
- Interaktiver Python-Shell fehlt Vervollständigung etc.

Nachteile

- unklare Lizenzsituation
- Kommunikation mit den Entwicklern schwierig
- Interaktiver Python-Shell fehlt Vervollständigung etc.
- Nutzung der integrierten Bibliotheken über die integrierte Python-Shell größtenteils nicht möglich

Nachteile

- unklare Lizenzsituation
- Kommunikation mit den Entwicklern schwierig
- Interaktiver Python-Shell fehlt Vervollständigung etc.
- Nutzung der integrierten Bibliotheken über die integrierte Python-Shell größtenteils nicht möglich
- Mangelhafte Paketverwaltung

API-Beispiel

```
1 import android
2 droid = android.Android()
3
4 droid.dialogCreateAlert("Test", "Bitte OK druecken.")
5 droid.dialogSetPositiveButtonText("OK")
6 droid.dialogShow()
7 result = droid.dialogGetResponse().result
8 droid.dialogDismiss()
```

Listing 1: OK-Dialog

API-Wrapper I - Stifte und Mäuse

Vorteile:

- Umsetzung von Stifte und Mäuse für Python existiert
- Viele Module und Materialien vorhanden
- (Theoretisch) plattformunabhängig

API-Wrapper I - Stifte und Mäuse

Vorteile:

- Umsetzung von Stifte und Mäuse für Python existiert
- Viele Module und Materialien vorhanden
- (Theoretisch) plattformunabhängig

Nachteile:

- Bekannte Probleme von SuM
- Umsetzung grafischer Bestandteile nicht unproblematisch
- Verwirrung der SuS bei unvollständiger Umsetzung

API-Wrapper II - Eigener Ansatz

Vorteile:

- Keine Verwirrung bei Kenntnis von SuM
- Für mobile Geräte optimierbar
- (Theoretisch) plattformunabhängig
- Parallele Nutzung von Attributen und Kapselung
- Anpassung an Vorstellungen der SuS möglich

API-Wrapper II - Eigener Ansatz

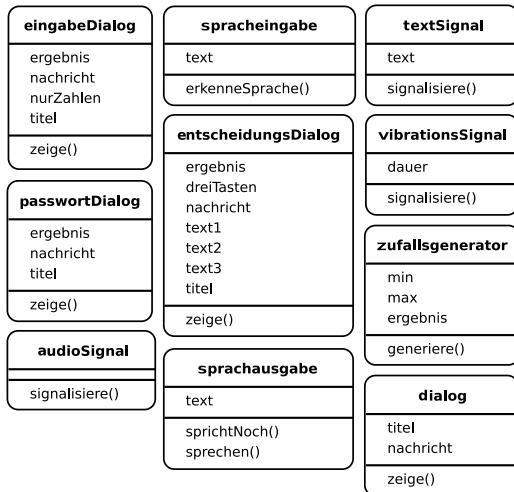
Vorteile:

- Keine Verwirrung bei Kenntnis von SuM
- Für mobile Geräte optimierbar
- (Theoretisch) plattformunabhängig
- Parallele Nutzung von Attributen und Kapselung
- Anpassung an Vorstellungen der SuS möglich

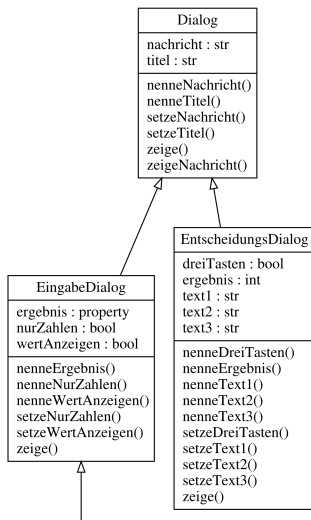
Nachteile:

- Mehraufwand
- Vielfache Implementierung erforderlich
- Bindung an bereits implementierte Plattformen

API-Wrapper III - Mögliche Ausgestaltung



API-Wrapper III - Aktuelle Umsetzung (Auszug)



API-Beispiel

```
1 from velamentum.alle import *
2
3 dialog = Dialog()
4
5 dialog.titel = "Test"
6 dialog.nachricht = "Bitte Ok druecken."
7
8 dialog.zeige()
```

Listing 2: OK-Dialog, objektorientiert

Vorteile

- Stärkerer Alltagsbezug
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

Vorteile

- **Stärkerer Alltagsbezug**
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

Vorteile

- Stärkerer Alltagsbezug
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

Vorteile

- Stärkerer Alltagsbezug
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

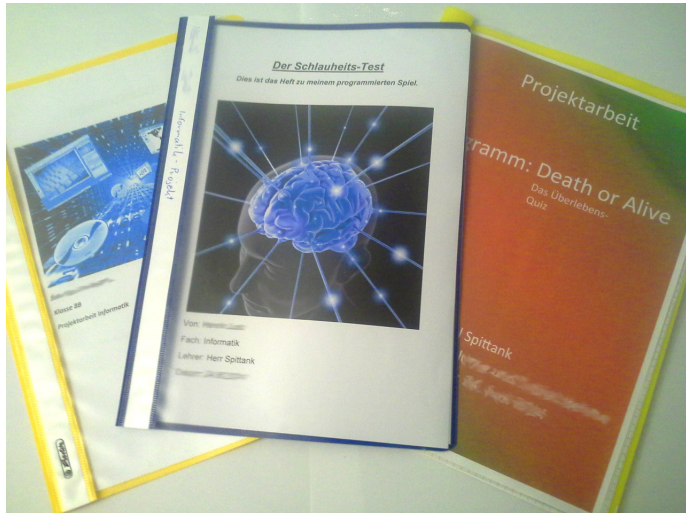
Vorteile

- Stärkerer Alltagsbezug
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

Vorteile

- Stärkerer Alltagsbezug
- Flexiblere Unterrichtsgestaltung
- Außerunterrichtliche Nutzung
- Mehr sachbezogene Kommunikation und Interaktion
- Höhere Motivation
- Nutzung eigener Geräte
 - Verringerter Wartungsaufwand
 - Geringere Kosten

Projekte



Projekte

Mal-Taschenrechner Programmablauf

Bevor man den Mal-Taschenrechner benutzt, sollte man sich erst einmal eine grobe Aufgabe überlegen haben. Über das Ansehen der Photo-Foren bei Python öffnen sich der Taschenrechner. Zuerst wird ein Inputfenster mit dem Text "Geben Sie die erste Zahl der Malaufgabe ein!!!".

Mal-Taschenrechner Anforderungen

Zu beachten sind:

- Der Benutzer hat sich vor dem Benutzen eines Mal-Aufgabe überlegt.
- Die Aufgabe und das Ergebnis kann unendlich lang sein.
- Es können keine Kommastellen, Brüche, etc. eingegeben werden.
- Es darf nur den Anforderungen des Programms folgen.

Mal-Taschenrechner Projektdiele

Ich bin auf die Idee gekommen einen Taschenrechner für Malaufgaben zu programmieren, weil im Alltag hauptsächlich Additions- und Multiplizieraufgaben vorkommen. Die anderen vier Rechenarten sind mir nicht so wichtig, deshalb habe ich mich dazu entschieden eine Hilfsfunktion für Malaufgaben zu programmieren.

Mal-Taschenrechner Python-Code

```

from vclamentum.alle import*
ed=InputDialog("Taschenrechner","Geben sie die erste Zahl der Malaufgabe ein!!!")
ed.nurZahlen=True
ed.zeige()
x=ed.ergebnis
ed=InputDialog("Taschenrechner","Geben sie die zweite Zahl der Malaufgabe ein!!!")
ed.nurZahlen=True
ed.zeige()
y=ed.ergebnis
z=(int(x)*int(y))
d=Dialog("Ergebnis",str(z))
d.zeige()
                    
```


2. Planung

kategorie

```

text = "Kategorie"
auswählen = true
fragen = 10
punkte = 10
                    
```

auswählen ()

mischacht ()

frage

```

text = "frage"
richtig/beantwortet = +1
                    
```

Diese Objektkarte gilt den Kategorien. Die Kategorien sind dazu da, um auszuwählen, was man bei diesem Quiz spielen möchte. Das Attribut 'text' wird mit dem Namen der Kategorie ausgestattet. In der Objektkarte steht je nach Kategorie, dort sollte aber so etwas wie beispielsweise 'League of Legends' stehen. Kategorien kann man auswählen, daher das Attribut 'auswählen'. Da es zehn Fragen sind insgesamt zehn Punkte gibt werden die Attribute 'fragen' und 'punkte' noch verwendet. Man kann die Kategorie entweder auswählen oder mischen, daher die Methoden 'auswählen()' und 'mischacht()'.

Diese Objektkarte gilt den Fragen. Die Fragen sind dazu da, um sie zu beantworten. Sie werden durch die Kategorie ausgewählt.

Nachteile

- Erschwerte Bedienung
- **Verringerte Geschwindigkeit**
- ⇒ Frustration?
- Exklusive Nutzung der Geräte

Nachteile

- Erschwerte Bedienung
- **Verringerte Geschwindigkeit**
- ⇒ Frustration?
- Exklusive Nutzung der Geräte
- **Irritation von SuS mit Programmiererfahrung**

Nachteile



- Erschwerte Bedienung
- **Verringerte Geschwindigkeit**
- ⇒ Frustration?
- Exklusive Nutzung der Geräte
- **Irritation von SuS mit Programmiererfahrung**
- **Update-Tücken**

Online-Material



<https://edu.spittank.net/downloads/mobile/>

Literatur I

-  Carrie, Ralph (2006). „Einsatz mobiler Informatiksysteme im Informatikunterricht der gymnasialen Oberstufe“. Hausarbeit gemäß OVP. Hamm: Studienseminar für Lehrämter an Schulen – Seminar für das Lehramt für Gymnasien Gesamtschulen. URL: <http://www.ham.nw.schule.de/pub/bscw.cgi/315319>.
-  Heming, Matthias (2009). „Einsatzszenarien von Mobiltelefonen im Informatikunterricht“. Masterarbeit – Master of Education. Wuppertal: Bergische Universität – Fachbereich Mathematik und Naturwissenschaften. URL: <http://blog.familie-heming.de/?p=111>.

Literatur II



MPFS (2013). *JIM 2013. Jugend, Information, (Multi-)Media. Basisuntersuchung zum Medienumgang 12- bis 19-Jähriger in Deutschland*. Forschungsbericht. MPFS – Medienpädagogischer Forschungsverbund Südwest. Stuttgart: mpfs. URL: <http://www.mpfs.de/fileadmin/JIM-pdf13/JIMStudie2013.pdf%E2%80%8E>.



Spittank, Daniel (2012). „Auswahl und Gestaltung mobiler Informatiksysteme für den Einsatz im Informatikunterricht“. Masterarbeit – Master of Education. Wuppertal: Bergische Universität – Fachbereich Mathematik und Naturwissenschaften. URL: <https://edu.spittank.net/downloads/mobile/examensarbeit.pdf>.