



Ist das Designen und Programmieren von Computerspielen für Kinder von Nutzen?

Eine Metasynthese der Forschungsergebnisse



Was ist „Computer Game Design & Programming“ (CGD&P)?

Unter Computer Game Design & Programming (CGD&P) versteht man Lernsettings, in denen Kinder eigene Computerspiele entwerfen und programmieren. Sie kombinieren Storytelling, Grafik, Logik und Interaktion, nutzen dabei kindgerechte Sprachen wie Scratch oder Kodu und arbeiten häufig in Teams.



Forschungssynthese

Problemstellung

- Programmieren und Gestalten von Computerspielen wird als vielversprechender Ansatz für das Lernen von Kindern gesehen
- Bisher gab es keine systematische Übersicht über empirische Belege für Nutzen in Bezug auf Programmieren, Problemlösen, Motivation und Haltung zur Informatik
- Unterschiedliche Tools, Zielgruppen und didaktische Ansätze erschweren die Einschätzung der Wirksamkeit

Ziel der Forschungssynthese

- Systematische Meta-Synthese von Studien (K-12) zu den Effekten des Designs und Programmierens von Computerspielen (CGD&P)
- Analyse, inwieweit es Belege für Verbesserungen in Programmierwissen, Problemlöse- und Denkfähigkeiten, Einstellung und Selbstvertrauen in Informatik gibt

Forschungsfragen:

Welche Lernergebnisse und motivationalen Effekte werden berichtet?
Wer profitiert am meisten (Alter, Geschlecht, Vorerfahrung)?
Unter welchen pädagogischen Bedingungen sind die Effekte am stärksten?
Wo bestehen Forschungslücken und methodische Schwächen?

Methodik

- Meta-Synthese qualitativer, quantitativer und Mixed-Methods-Studien
- Literaturrecherche (Datenbanken, Konferenzbände, NSF-Projekte, Autorenanfragen) → >400 Treffer
- Einschlusskriterien: Schule, Spiele selbst programmieren (nicht nur gestalten), klare Daten zu Lernen/Motivation
- 68 Studien (2000–2015) in Analyse aufgenommen

Ergebnisse

Programmieren & Fachwissen:

- Zahlreiche Studien belegen Zuwächse im Programmierwissen (z. B. Sequenzen, Bedingungen, Debugging)
- Auch Fähigkeiten wie Algorithmisches Denken, Logik und Storytelling verbessern sich
- Positive Effekte treten sowohl bei strukturierten Curricula als auch bei explorativen Ansätzen auf

Problemlösen & Computational Thinking:

- Hinweise auf bessere Planung, Testen und iterative Verbesserung von Lösungen
- Pair-Programming oder Teamarbeit unterstützen strategisches Denken

Einstellungen & Motivation:

- Viele Kinder berichten gesteigertes Interesse an Informatik und mehr Selbstvertrauen
- Einzelne Studien zeigen sinkende Motivation, wenn Aufgaben zu komplex oder unklar strukturiert sind
- Mädchen und marginalisierte Gruppen profitieren besonders, wenn soziale Interaktion und kreative Freiheit geboten werden

Pädagogische Faktoren:

- Effekte hängen stärker von Unterrichtsgestaltung und Reflexionsphasen ab als vom gewählten Tool
- Kombination aus Anleitung, Feedback und Peer-Austausch besonders förderlich

Einordnung DigComp

Umgang mit Informationen & Daten

Kommunikation & Zusammenarbeit

Kreation digitaler Inhalte

Sicherheit

Problemlösen

Praktische Implikationen

- Game-Design-und-Programmierprojekte sind **wertvolle Lernumgebungen** für Informatik- und Problemlösekompetenzen
- Erfolgreich, wenn **klare Ziele**, strukturierte Einführung, **soziale Lernformen** und **kreative Gestaltungsfreiräume** kombiniert werden
- Sinnvoll **besonders für jüngere Lernende und Mädchen**, um Interesse an MINT zu fördern
- Lehrkräfte benötigen didaktische Unterstützung, um Reflexion und iterative Entwicklungsprozesse zu begleiten

Verbundpartner des Metavorhabens: