

FAKULTÄT FÜR INFORMATIK

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatik

Spielend zum Lernerfolg?

**Einfluss und Wirkung des Einsatzes einer 2D-Spiele-Engine auf den
gymnasialen Informatikunterricht in Wechselwirkung mit dem bayerischen
Lehrplan**

Michael Andonie

FAKULTÄT FÜR INFORMATIK

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatik

Spielend zum Lernerfolg?

**Einfluss und Wirkung des Einsatzes einer 2D-Spiele-Engine auf den
gymnasialen Informatikunterricht in Wechselwirkung mit dem bayerischen
Lehrplan**

Learning with fun and games?

**Influence and effects of using a 2D-game-engine in computer science class
and interdependencies with the curriculum of the bavarian secondary
education**

Author: Michael Andonie
Supervisor: Gudrun Johanna Klinker, Prof. Ph.D.
Advisor: Sandro Weber, M.Sc.
Submission Date: 15.12.2016

Ich versichere, dass ich diese Bachelor's Thesis selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, 15.12.2016

Michael Andonie

Danksagungen

Diese Arbeit wäre nicht möglich geworden ohne die konstante, enthusiastische und aufopferungsvolle Unterstützung von *OStR Michael Ganshorn*. Seine aktiven Beiträge zur Planung und Umsetzung von Erhebungen mit Schülern sind von unschätzbarem Wert für diese Arbeit.

Ich möchte mich bei meinem Betreuer *Sandro Weber* bedanken. Seine Unterstützung und der enorme Freiraum, den er mir bei der Wahl und Bearbeitung dieses besonderen Themas bereitstellte, machten diese besondere Arbeit - mit unüblichem Themenbereich und unüblicher Bearbeitungszeit - erst möglich.

Abstract

Contents

Danksagungen	iii
Abstract	iv
1 Einleitung und Motivation	1
2 Die Engine Alpha	2
3 Entwicklung der Thesen	4
3.1 Der Lehrplan am bayerischen Gymnasium	4
3.1.1 Bevorstehende Lehrplanänderungen am Gymnasium	4
3.1.2 Konflikte beim Lehrplan PLUS für Gymnasium	5
3.2 Konkrete Konfliktthemen dieser Arbeit	5
3.2.1 Motivation durch Sachverständnis vs. Motivation durch Zurückhaltung komplexer Inhalte	5
3.2.2 Lernergebnisse durch pragmatische Aufgaben vs. Lernergebnisse durch spielerische Aufgaben	6
3.2.3 Verständnis durch Zurückhaltung von Vererbung vs. Verständnis durch Einführung von Vererbung	7
4 Entwurf der Tests	8
4.1 Allgemeine Umsetzung von Schülertests	8
4.2 Konzipierung der Schülertests	8
4.2.1 Testlauf 1: Unterrichtsvergleich (Engine vs. Standard-Unterricht)	9
4.2.2 Testlauf 2: Parallele Unterrichtssequenzen	9
4.2.3 Testlauf 3: Umfrage	12
4.3 Umsetzung der Testläufe	12
5 Ergebnisse und Einsichten aus den Tests	15
5.1 Einsichten aus Testlauf 1	15
5.1.1 Der Umstieg auf die Engine steigert Motivation und Selbstwirksamkeit	15
5.1.2 Kritische Hinterfragung der vorigen Schlussfolgerung	16

Contents

5.1.3	Der Unterricht mit der Engine verbessert die Selbsteinschätzung und das Fachverständnis der Schüler	19
5.1.4	Der Unterricht mit der Engine verbessert die wahrgenommene Unterrichtsqualität	19
5.2	Einsichten aus Testlauf 2	20
5.2.1	Je spielerischer das Projekt, desto beliebter und motivierender ist es bei den Schülern	21
5.2.2	Eine verpflichtende Erstausswahl verändert den Bezug und die Motivation zum Unterricht	21
5.2.3	Wechselnde Schüler erzielen bessere Testergebnisse als bleibende	22
5.2.4	Wo vergleichbar,	22
6	Verifikation und Diskussion der Thesen	25
6.1	Verifikation und Diskussion zu These 1	25
6.2	Verifikation und Diskussion zu These 2	25
6.3	Grenzen der Verifikation	26
7	Zukünftige Schritte im Zuge dieser Arbeit	27
8	Abschließende Bewertung des Lehrplans im Zuge der Ergebnisse	28
9	Abschließende Worte	29
	List of Figures	30
	List of Tables	31
	Bibliography	32

1 Einleitung und Motivation

Education is the most powerful
weapon which you can use to change
the world.

Nelson Mandela

Lernen und Lernprozesse begleiteten mich bewusst seit mehreren Jahren. Mich treibt die Überzeugung, dass eine gute Ausbildung – innerhalb und außerhalb der Schule – den Erfolg von Menschen und die Zukunft unserer Gesellschaft bestimmt. Aus diesem Grund arbeite ich seit Jahren an verschiedenen Bildungsprojekten innerhalb und außerhalb Deutschlands; technologisch und didaktisch. Eines dieser Projekte hat mich dazu bewegt, diese Arbeit zu schreiben.

Das 2010 von mir gestartete Projekt *Engine Alpha* und dessen Einsatz im Informatikunterricht ist zentraler Gegenstand dieser Arbeit. Seit über 6 Jahren arbeite Ich an der Entwicklung, didaktischen Konzepten und Organisation der Engine Alpha Software, einer 2D Game Engine. Hierbei konnte ich den immensen Einfluss erleben, den die Entwicklung von Computerspielen auf Schüler im Informatikunterricht hat. Mit dieser Arbeit möchte ich diese persönlichen Wahrnehmungen quantifizieren und die Ergebnisse als Grundlage nutzen, die Engine zu popularisieren und weiterzuentwickeln.

Diese Bachelorarbeit beschäftigt sich mit menschlichem Lernen, Motivation und deren Umsetzung in der Schule. Mit den *bevorstehenden Lehrplanänderungen* im bayerischen Gymnasium können die Ergebnisse dieser Arbeit auch relevante, quantifizierbare Einsichten schaffen, um die neuen Pläne möglichst effektiv und einflussreich für die Lernprozesse der Schüler zu gestalten – und damit deren Erfolg auch außerhalb der Schule.

Hierfür wurden *Evaluationen durch Lehrer* angestellt, deren zentrale Fragestellungen an *aktuellen Streitpunkten im neuen Lehrplan* angelehnt sind, um hierfür relevante Ergebnisse zu erreichen. Diese können anschließend in die aktuellen Diskussionen zurückfließen, und so pragmatische Impulse für die Gespräche liefern.

Damit ist der Zweck dieser Arbeit, einen *hilfreichen Beitrag* für die Lehrplangestaltung zu liefern. Aus diesem Grunde wurde *Deutsch als Sprache* für diese Arbeit gewählt.

2 Die Engine Alpha

Diese Arbeit beschäftigt sich damit, wie Schüler am erfolgreichsten im Informatikunterricht lernen können. Hierfür werden verschiedene *Lehrmethoden und -paradigmen* betrachtet. Diese Arbeit hat die *Engine Alpha* als Unterstützungsmittel für Informatikunterricht im Fokus. Zunächst sei diese hier umrissen.

Die Engine Alpha ist eine Java-basierte 2D Open-Source Game Engine und wurde ursprünglich entwickelt von mir.[[ea_website](#)] Ihr Fokus liegt nicht darin, eine plattformübergreifende, effektive und kommerziell interessante Spieleentwicklung zu ermöglichen. Stattdessen liegt ihr Fokus auf der *Verwendung durch Schüler*. Damit sind einige zentrale Design-Entscheidungen entgegen üblicher Konventionen für professionelle Spiele-Engines getroffen worden. Die wichtigsten sind hierbei:

- *Deutsche API-Sprache:*

Sämtliche Schnittstellen, mit denen ein Nutzer in Berührung kommt, sind in Deutsch verfasst. Die reduzierte Sprachbarriere ermöglicht Schülern einen leichteren Umgang mit dem Framework.[Enga]

- *Orientierung am Lehrplan:*

Unterrichtsinhalte werden deutlich und anschaulich durch die Engine-API so eingeführt und verwendet, dass der Umgang mit der Engine für Schüler möglichst intuitiv und einfach ist; sie finden „altbekannte“ Vorgehensweise, Konventionen und Konzepte wieder bei der Verwendung der Engine.[Engb]

- *Umfassende Tutorials und Codebeispiele:*

Eine große Sammlung von Schritt-für-Schritt-Anleitungen und Beispielprojekte verschiedener Größe ermöglichen einen guten Einblick in die korrekte Nutzung der Engine. Diese sind in Zusammenarbeit mit Lehrkräften entstanden und orientieren sich damit stark am Vokabular und den Strukturen, die Schüler bereits durch den Unterricht gewohnt sind. [Engc]

- *Reduktion aller Konzepte wo möglich:*

Professionelle Spieleentwicklung bedarf in der Regel umfassenden Wissens um verschiedene technische Konzepte, von frameweiser Abarbeitung über Shader

und Collision Detection bis hin zu I/O-Operationen. Auch auf Schnittstellenebene entsteht so ein Overhead, den man einem Schüler am Beginn seiner Programmiererfahrung kaum zumuten kann.

Die Engine kapselt nahezu all diese Standardoperationen, sodass Schüler sie unmittelbar nutzen, ohne sich der hinterliegenden Konfigurationen bewusst zu sein.

Mit dem hinterliegenden Grundsatz: „Möglichst barrierefreier Start für Anfänger, sonst übersichtliche und optionale Features“ bietet sich die Engine besonders für Schüler an und ermöglicht einen pragmatischen, anwendungsbezogenen Rahmen für die Unterrichtsgestaltung. Seit der ersten Veröffentlichung der Engine 2011 hat sich durch Zusammenarbeit mit Informatiklehrern eine Sammlung an Unterrichtssequenzen und -methoden basierend auf der Engine gebildet ([Engb]).

Besonders die absolut minimalistische EDU-Variante der Engine findet durch eine breite Aufstellung von darauf basierenden Unterrichtssequenzen und enormer Barrierefreiheit für Schüler viel Anwendung im Informatikunterricht.

Seit Start des Projekts ist um die Engine ein Kreis an Beitragenden entstanden – auf Entwicklungs- und auf Didaktikseite. Die Engine wird in ihrer aktuellen Veröffentlichung für das Gymnasium (Version 3.x) gewartet; die Entwicklung einer überholten und optimierten Version 4.0 läuft derzeit.[[ea_changelog](#)]

3 Entwicklung der Thesen

Die Engine Alpha ist Ausgangspunkt dieser Arbeit. Mit dem Überblick über deren Rolle im Informatikunterricht lässt sich nun die wissenschaftliche Arbeit um die Engine und den Lehrplan betrachten. Hierfür sei zunächst *der Hintergrund und die Relevanz der Arbeit* umrissen; diese liegen in den aktuellen Entwicklungen im bayerischen Lehrplan.

3.1 Der Lehrplan am bayerischen Gymnasium

Der Lehrplan ist das inhaltlich wohl relevanteste Dokument für die schulische Lehre in Deutschland. Dieser ist für Informatik in seiner aktuellen Form seit 2004 gültig (in Fassung von 2009)[Kulc] und wurde im Schuljahr 2007/2008 zum ersten Mal in der 10. Klasse umgesetzt. Im folgenden wird der Lehrplan der 10. Klasse Informatik am bayerischen Gymnasium als zentraler Gegenstand dieser Arbeit betrachtet und abgekürzt als Lehrplan bezeichnet.

3.1.1 Bevorstehende Lehrplanänderungen am Gymnasium

Seit 2013 wird an der Umsetzung einer schulartenübergreifenden ([Kula] Lehrplanänderung gearbeitet, dem *Lehrplan PLUS*. In den Grundschulen ist dieser bereits in Kraft getreten ([BLV]); weitere Einführungen folgen. ([Kula], auch im folgenden)

Die prominenteste Änderung durch den neuen Lehrplan ist die *Kompetenzorientierung*: Lernziele werden durch Kompetenzerwerb beschrieben (sowie weiterhin durch Wissenserwerb, welcher sekundär stets mitgeführt wird).

Mit dem Paradigmenwechsel werden auch die Inhalte der Lehrpläne überarbeitet. Für die Oberstufe Informatik am Gymnasium sind die geplanten inhaltlichen Veränderungen durchaus nennenswert: (vgl. [Stac] und Unterseiten mit [Kulb] und [Stab])

- *Verlagerung von Vererbung von der 10. in die 11. Klasse:*

Die objektorientierte Programmierung per se bleibt weiter Gegenstand der 10. Klasse. Jedoch wird Vererbung als wichtiges Konzept der OOP vom Lehrplan der 10. Klasse ausgelagert; als fakultatives Thema ist es zum Schuljahresende am Rande enthalten. Dafür wird die Vererbung als verpflichtender Inhalt zu Beginn der 11. Jahrgangsstufe eingeführt.

- *Modellierung gewinnt an Bedeutung auf Kosten von Implementierung*

Wesentliche Konzepte und Inhalte des Lehrplans bleiben bestehen. Der Fokus hierbei verschiebt sich jedoch von Implementierung hin zu Modellierung. So müssen Konzepte wie Synchronisation paralleler Prozesse oder Netzwerke in Modellierung (durch UML oder Petrinetze) bearbeitet werden. Deren Implementierung jedoch wird nunmehr fakultativ und damit nicht mehr prüfungsrelevant.

3.1.2 Konflikte beim Lehrplan PLUS für Gymnasium

Diese Veränderungen wurden nicht durchweg positiv aufgenommen: Bei den zugehörigen Fachlehrertagungen hat sich eine "beachtliche Unzufriedenheit" gegenüber dem neuen Konzept gebildet: Eine überwiegende Mehrheit von verantwortlichen Seminarlehrern und Fachreferenten zeigte sich nicht zufrieden mit den Lehrplanänderungen; auch von Hochschuldidaktikern kamen Bedenken.[Gan16]

Die Konflikte sind nicht unbedingt ein Zeichen für Schwäche in der Planung. Sie können ein äußerst hilfreicher Antrieb für die Erstellung eines optimalen Lehrprozesses sein, der auf die Lernerfahrung und die Lernergebnisse der Schüler einen möglichst positiven Einfluss nimmt. In diesem Sinne richtet sich diese Arbeit an Konfliktthemen zum Lehrplan mit der *Ambition, quantifizierbare, fundierte und sachorientierte Ergebnisse* für die Optimierung des Schulunterrichts zu bringen.

Hierbei ist nicht zu leugnen, dass durch die Engine als Fokus der Arbeit bereits eine vorgefestigte Meinung beim Autor vorliegt. Als Autor bin ich mir dieses Bias bewusst und bin bestens bemüht, diesem mit neutraler, wissenschaftlicher Arbeit zu begegnen; letztlich trägt auch der Review-Prozess dieser Arbeit dazu bei, dass die Ergebnisse abschließend valide sind, unabhängig von der Meinung des Autors.

3.2 Konkrete Konfliktthemen dieser Arbeit

Die Meinungen zur Lehrplangestaltung sind vielseitig und komplex. Natürlich kann diese Arbeit in ihrem angedachten Umfang nicht alle aktuellen Problematiken abdecken. Stattdessen fokussiert sie die Arbeit auf *drei aktuelle Konfliktthemen*, die einerseits in den Diskussionen relevant sind und andererseits im Bezug auf die Engine und deren Verwendung im Unterricht interessant sind.

3.2.1 Motivation durch Sachverständnis vs. Motivation durch Zurückhaltung komplexer Inhalte

Schülermotivation ist ein elementarer Bestandteil der Lehrplangestaltung. Die Motivation eines Lernenden bestimmt im Allgemeinen die Bereitschaft, sich bewusst mit

Inhalten auseinanderzusetzen. Hierbei scheiden sich derzeit die Geister in einem wesentlichen Punkt. Die Frage, *welche Lerninhalte die Schüler motivieren*, wird sehr unterschiedlich gesehen: Es gibt Vertreter, die argumentieren, dass umfassendes Sachverständnis und Hintergrundwissen die Schüler motiviert, zum Beispiel durch fakultative und vertiefende Lerninhalte. Andere argumentieren, dass das Gegenteil zutreffe: Anstatt die Schüler mit vertiefenden Lerninhalten möglicherweise zu überfordern, solle man eher daran arbeiten, nicht unmittelbar relevante Lerninhalte zurückzuhalten, um so die zentralen Lerninhalte im Fokus zu halten und die Schüler nicht womöglich zu überlasten.

Die Engine Alpha folgt gemäß ihres Designs der zweiten Überzeugung. Nicht unmittelbar relevante Konzepte (z.B. GUI-Design oder Computergrafik) werden komplett zurückgehalten, sodass die Nutzung der Engine ohne zusätzliches Sachverständnis (über die ohnehin notwendigen Grundlagen der Programmierung) möglich ist. Dahinter liegt unabstreitbar die Überzeugung, dass dieser Ansatz besser sei als die Alternative. Unabhängig von diesem Bias soll diese Fragestellung natürlich wissenschaftlich neutral beantwortet werden. Um dies also quantifizierbar und möglichst wertfrei zu erfassen, wird diese Haltung als zu verifizierende These verfasst:

These 1

Bei Programmieraufgaben fördert *Simplizität und Rückhaltung* komplexer und (momentan) nichtrelevanter Lerninhalte in der Aufgabenstellung mehr *Kreativität* und damit *Motivation* bei den Schülern als ein *umfassendes Aufgabenspektrum*, das ganzheitliches Hintergrundverständnis entwickelt.

3.2.2 Lernergebnisse durch pragmatische Aufgaben vs. Lernergebnisse durch spielerische Aufgaben

Motivation ist nicht der einzige Faktor, den der Lehrplan möglichst weit entwickeln möchte. Der primäre Fokus eines Lehrplans ist natürlich das Lernen. Lerninhalte sollen möglichst schnell und umfassend verstanden werden. Auf die Frage, wie genau das passieren soll, also welche Lehrmethodik zu bevorzugen ist, gibt es viele Meinungen; und erneut gegensätzliche Ansichten. So finden sich Verantwortliche, die pragmatische Aufgaben und Anwendungen „mitten aus dem Leben“ bevorzugen, wie z.B. eine Aufgabe, bei der ein Bankkonto objektorientiert modelliert und umgesetzt wird. Genauso finden sich Ansichten, dass durch spielerische, weniger pragmatische Aufgaben die Lernleistung von Schülern höher sei; z.B. in dem ein Computerspiel programmiert wird.

Die verschiedenen Philosophien zur Umsetzung des Unterrichts finden sich bereits seit Einführung des Lehrplans (vgl. Konzept 1 und Konzept 2 von [Elk08]).

Die Engine Alpha vertritt auch hier konzeptuell eine klare und offensichtliche Position: Spielerische, nicht unbedingt pragmatische Aufgaben motivieren Schüler mehr und steigern so die Bereitschaft, sich mit den Lerninhalten auseinanderzusetzen und so besser Leistungen zu erzielen. Erneut wird diese Position im Sinne fairer wissenschaftlicher Arbeit als wertende These formuliert, die neutral verifiziert bzw. falsifiziert werden soll:

These 2

Die *spielerische Komponente* bei Aufgaben führt zu mehr *Motivation und Lernzuwachs* im Informatikunterricht als die *pragmatisch anwendungsbezogene Komponente*.

3.2.3 Verständnis durch Zurückhaltung von Vererbung vs. Verständnis durch Einführung von Vererbung

Der letzte Konfliktpunkt, dessen Evaluation für diese Arbeit angedacht ist, ist eine vergleichsweise spezifische Inhaltsfrage: Es geht um das Konzept der objektorientierten Vererbung. Diese ist bisher fester Teil des Lehrplans. Nun gibt es die Möglichkeit, diesen Inhalt mit der kommenden Lehrplanerneuerung komplett zu streichen ([Staa], Lernbereich 4). Diese Option ist durchaus umstritten, mit Befürwortern und Gegnern auf beiden Seiten ([Gan16]).

Die Engine lässt sich auch hier klar einem der „Lager“ zuordnen. Vererbung ist ein elementarer Aspekt der API-Nutzung der normalen Engine¹. Erfahrungsgemäß unterstützt die Einführung von Vererbung durch die Engine nicht nur das Verständnis für das Konzept selbst, sondern führt auch allgemein zu tieferem Verständnis für die Sinnhaftigkeit der objektorientierten Programmierung. Erneut wird diese Position als These formuliert:

These 3

Die *frühzeitige Einführung der Vererbung* im Informatikunterricht führt zu *deutlich mehr Verständnis* für die Sinnhaftigkeit des Konzeptes sowie für die objektorientierte Modellierung/Programmierung an sich.

¹Die weiter reduzierte EDU-Variante benötigt Vererbung auf Anwenderseite nicht unbedingt

4 Entwurf der Tests

Anhand aktueller Fragen zum Lehrplan wurden nun Thesen entworfen, die die Tauglichkeit der Engine für den Informatikunterricht auf den Prüfstand stellen. Im Folgenden können darauf basierend konkrete, experimentelle Tests erstellt werden, die Einsichten hierzu ermöglichen.

4.1 Allgemeine Umsetzung von Schülertests

Sämtliche Thesen beziehen sich auf Schüler des bayerischen Gymnasiums; die Interaktion mit ihnen zur Erhebung von relevanten Daten ist also nicht vermeidbar. Arbeit mit Schülern ist jedoch nicht ohne Weiteres durchführbar. Strenge Beschränkungen sichern den Datenschutz der zumeist minderjährigen Lernenden. Für einen Studenten der Informatik ist der direkte Zugang hier kaum möglich.

Um die Datenerhebung mit Schülern im weiteren Sinne zu ermöglichen, wurde diese Arbeit durch *OStR Michael Ganshorn* extern betreut, Seminarlehrer für Informatik am ASG Passau und beteiligter an den Lehrplandiskussionen. Die vom Autor konzipierten Datenerhebungen wurden von ihm praktisch in die Wege geleitet. Michael Ganshorn ist ebenfalls aktiv am Engine Alpha Projekt beteiligt. Er arbeitet seit mehreren Jahren an didaktischen Konzepten und Unterrichtssequenzen basierend auf dem Framework. Natürlich stellen diese Aktivitäten einen möglichen Bias bei der Bearbeitung dar. Um die Gefahr eines solchen Einflusses auf die Testläufe zu minimieren, bestand konstante Rücksprache zwischen Michael Ganshorn und weiteren beteiligten Lehrkräften. Die geplanten Testläufe sind durch mindestens drei Instanzen gelaufen, bevor sie durchgeführt wurden: Michael Ganshorn, Michael Andonie sowie dritte bayerische Lehrkräfte.

4.2 Konzipierung der Schülertests

In Zusammenarbeit mit Michael Ganshorn wurden *drei konkrete Erhebungen* unterschiedlich in Größe, Fokus und Umfang konzipiert; jeweils eine Erhebung zur Erkundung einer der Thesen. Zwei der Erhebungen wurden bisher umgesetzt. Alle Drei Erhebungen werden im folgenden umrissen aus Sicht der Konzeption und bisheriger

Umsetzung. Sämtliche Erhebungen wurden für *Schüler der 10. Klasse Informatik an verschiedenen Gymnasien in Bayern* konzipiert und durchgeführt.

4.2.1 Testlauf 1: Unterrichtsvergleich (Engine vs. Standard-Unterricht)

Der erste angesetzte Testlauf orientiert sich an der ersten These. Diese beschäftigt sich mit der *Auswirkung des Umfangs der Unterrichtsinhalte auf die Schülermotivation*: die Frage, ob Zurückhaltung (momentan) irrelevanter Inhalte mehr Motivation als tiefgreifenderes Sachverständnis bringt, ist hiermit zu klären.

Um dies zu tun, wurde ein *Unterrichtsvergleich* konzipiert: Unterricht basierend auf Konzepten mit der Engine soll verglichen werden mit Unterricht, wie er in ISB-Handreichung oder Schulbüchern bisher beschrieben wurde.

Hierbei nimmt bei jedem Testlauf jeweils eine Schulklasse Teil, die bisher nicht mit der Engine unterrichtet wurde. Stattdessen hatte sie bisher „üblichen“ Unterricht. Diese Klasse macht jeweils einen Eingangstest. Hier wird neben Fachwissen vor allem wahrgenommene kreative Freiheit und Motivation erhoben. Nach dem Test wird die Arbeit mit der Engine Alpha eingeführt und eine Unterrichtssequenz damit durchgeführt. Danach wird abschließend ein Ausgangstest durchgeführt. Hierbei werden die exakt selben Daten wie zum Eingang erhoben. Die Datensätze können nun verglichen werden. Ein signifikanter Anstieg von Motivation bzw. wahrgenommener kreativer Freiheit nach dem Umstieg kann zur Untermauerung der ersten These beitragen. Zur Absicherung wird außerdem bei anderen Schulklassen zur gleichen Stelle im Lehrplan der Eingangstest ohne weiteres durchgeführt. Diese Klassen werden entweder bereits mit der Engine unterrichtet oder werden es im Laufe des Schuljahres gar nicht.

Die Unterrichtssequenz mit der Engine wurde von *Michael Ganshorn* in Rücksprache und Beratung mit dem Autor entwickelt.

4.2.2 Testlauf 2: Parallele Unterrichtssequenzen

Der zweite Testlauf basiert auf der zweiten These. Hierbei ist der Kontrast zwischen *spielerischen und pragmatischen Komponenten im Unterricht* im Fokus; es ist zu klären, wie dieser sich auf Motivation und Lernleistung auswirkt.

Hierzu wurden *parallele Unterrichtssequenzen* entwickelt. Als Lerninhalt wurde Programmierung mit Arrays gewählt; ein fester Bestandteil des aktuellen Lehrplans ([Kulb]) sowie in Lehrplan PLUS ([Staa]). Zu diesem Inhalt werden die parallelen Unterrichtssequenzen verglichen, die jeweils selbstständig von den Schülern bearbeitet werden.

Hierbei nimmt jeweils bei einem Testlauf eine Schulklasse Teil. Zu Beginn wird die Schulklasse auf die selbstständige Arbeit vorbereitet. Anschließend kann jeder Schüler

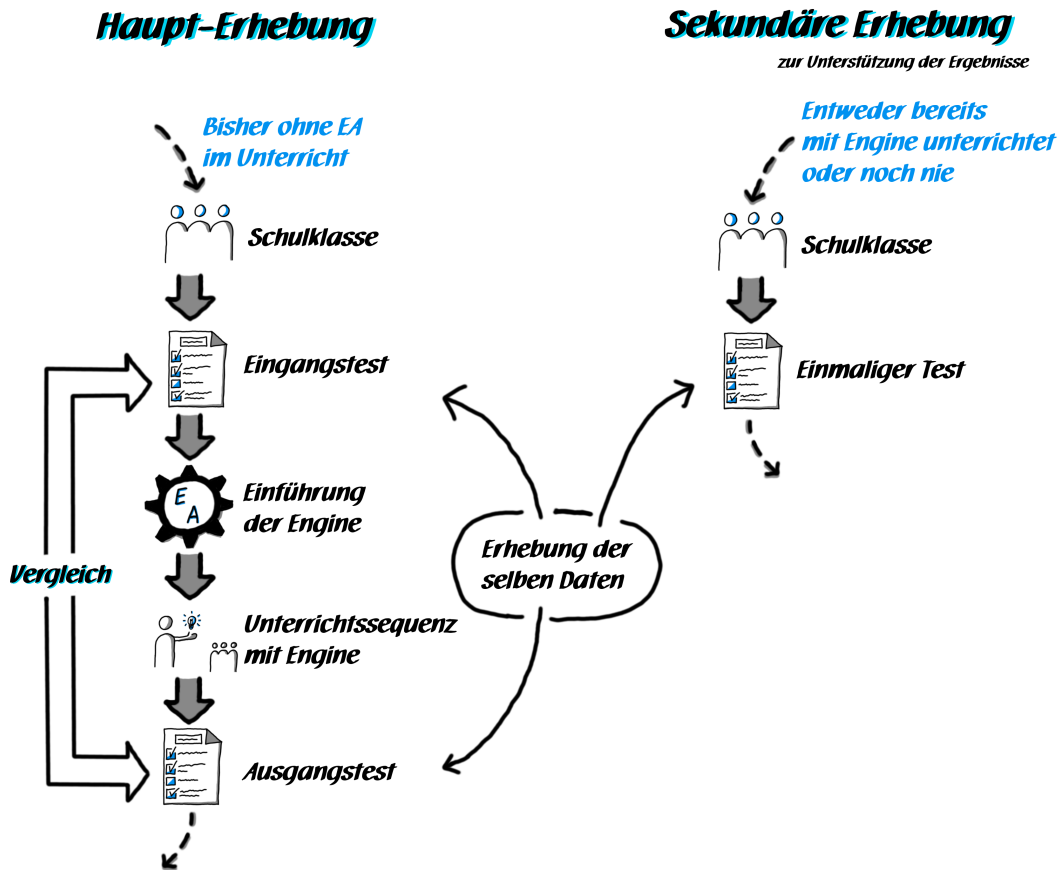


Figure 4.1: Visualisierung des Ablaufs der ersten Testreihe

für sich entscheiden, welche Sequenz mit identischem Lerninhalt er bearbeiten möchte. Zur Wahl steht:

- ein *pragmatisches Projekt* (eine Notenverwaltung mit einem einfachen GUI Tool)
- ein *spielerisches Projekt* (Entwicklung eines Jump n' Run Spiel mit der Engine)
- ein „*Middle Ground*“ *Projekt* (Programmierung einer Lichterkette mit der Engine als Backend)

Alle Projekte bearbeiten die Schüler selbstständig; sie werden vom Lehrer bei Rückfragen und Problemen betreut. Zur Selbstarbeit erhalten sie Skripte, anhand derer sie selbstständig ihr Projekt erarbeiten und sich das nötige Wissen aneignen.

Die Schüler bearbeiten im Folgenden die Sequenz. Nach der Hälfte der Bearbeitungszeit (2 Wochen) wird die Selbstarbeit pausiert und die Schüler tauschen sich in einer kurzen Veranstaltung über ihre gewählten Projekte und Erfolge aus. Anschließend beginnt die zweite Hälfte der Sequenzen in Selbstarbeit. Davor haben die Schüler die Möglichkeit, das Projekt zu wechseln und die zweite Hälfte mit einer der anderen drei Sequenzen anzugehen. Nach der zweiten Hälfte der Selbstarbeit wird eine abschließende Erhebung durchgeführt, bei der die Lernleistung durch einen fachlichen Test zu den Lerninhalten geprüft wird und die Schüler ihre autonomen Entscheidungen (wie Projektwahl und Projektwechsel) und Motivation darlegen können. Signifikante Unterschiede in Motivation und Lernleistung zwischen den verschiedenen Projektsequenzen und das begründete Wahlverhalten der Schüler lässt Rückschlüsse auf den Einfluss von pragmatischen und spielerischen Komponenten machen; fallen diese Daten zu Gunsten des Spielerischen Projektes aus, kann dies zur Untermauerung der zweiten These genutzt werden.

Zusätzlich nimmt eine *Ausnahmeklasse* an diesem Testlauf teil. Die wesentlichen Unterschiede hierbei sind:

- In dieser *beginnt jeder Schüler verpflichtend mit dem spielerischen Projekt*.
- Statt eines Projektaustausches nach der Hälfte findet eine *kurze Vorstellung der alternativen Projekte* statt.

Alle drei Unterrichtssequenzen wurden von *Michael Ganshorn* in Rücksprache und Beratung mit dem Autor entwickelt. Es wurde besondere Sorgfalt und Rücksicht genommen, um sicher zu stellen, dass alle Projekte dieselben Inhalte vermitteln, qualitativ gleichwertig sind, kreativ formuliert sind (mit offenen Entwicklungsmöglichkeiten für Schüler), und anschaulich aufgebaut sind. Besonders ist hierbei hervorzuheben, dass *alle Projekte auf das Wesentliche reduziert* sind, also kein weiteres Hintergrundwissen

von Schülern verlangt wird. Das so in jedem Fall reduzierte Aufgabenspektrum stellt sicher, dass es keine Überlagerung mit der ersten These und dem ersten Testlauf gibt.

Alle Skripte sind *im digitalen Anhang der Arbeit* enthalten.

4.2.3 Testlauf 3: Umfrage

Der dritte Testlauf basiert auf der dritten These. Der Fokus ist hier der *Zeitpunkt im Schuljahr, ab der Lerninhalt Vererbung eingeführt wird* und wie dies das Verständnis für die Programmierung beeinflusst. Dies ist – im Vergleich zu den vorigen Thesen – wesentlich enger gefasst. Damit lässt sich auch eine weniger Komplexe Erhebung gestalten.

In einer *einmaligen Umfrage* zum Schuljahresende können teilnehmende Schüler angeben, wann und wie in ihrem Unterricht Vererbung eingeführt wurde. Dazu werden Fragen zu Motivation, Lernleistung und Selbsteinschätzung (zur Programmierung) gestellt. Mit den so erhobenen Daten lässt sich die Einführung von Vererbung dem Verständnis für objektorientierte Programmierung gegenüberstellen. Signifikante Unterschiede im Verständnis abhängig von der Art der Einführung der Vererbung, können für die Diskussion der These genutzt werden.

Die Umfrage wurde vom Autor in Rücksprache mit Michael Ganshorn gestaltet.

4.3 Umsetzung der Testläufe

Sämtliche Tests sind nun umrissen. Bisher wurden *Testlauf 1 und Testlauf 2* durchgeführt. Sie wurden im Schuljahr 2015/16 an mehreren Gymnasien in Bayern jeweils in einzelnen Klassen nach vorgegebenem Schema ausgeführt. Dies ist jeweils in Form von Evaluationen durch die Klassenlehrer passiert. Michael Ganshorn hat diese Durchführung an Schulen betreut und organisiert.

An Testlauf 1 haben insgesamt *11 Schulklassen an 5 verschiedenen Schulen in Bayern* teilgenommen. Hierbei wurden *von Februar bis Mai 2016* über *330 Datensätze* erhoben.

An Testlauf 2 haben im *Mai und Juni 2016* insgesamt *3 Schulklassen* und damit insgesamt *62 Schüler* teilgenommen. Alle Klassen wurden – direkt oder indirekt – von *Michael Ganshorn* am ASG Passau betreut.

Testlauf 3 ist für das *Schuljahresende 2017* angesetzt. Da hier nicht klassenweise durch die jeweiligen Lehrer, sondern zentralisiert Daten erhoben werden, ist größerer Formaler Aufwand nötig. Nachdem Daten hier also nach Zeitpunkt der Veröffentlichung dieser Arbeit erhoben werden, wird dieser Testlauf im weiteren Verlauf dieser Arbeit nicht erwähnt.

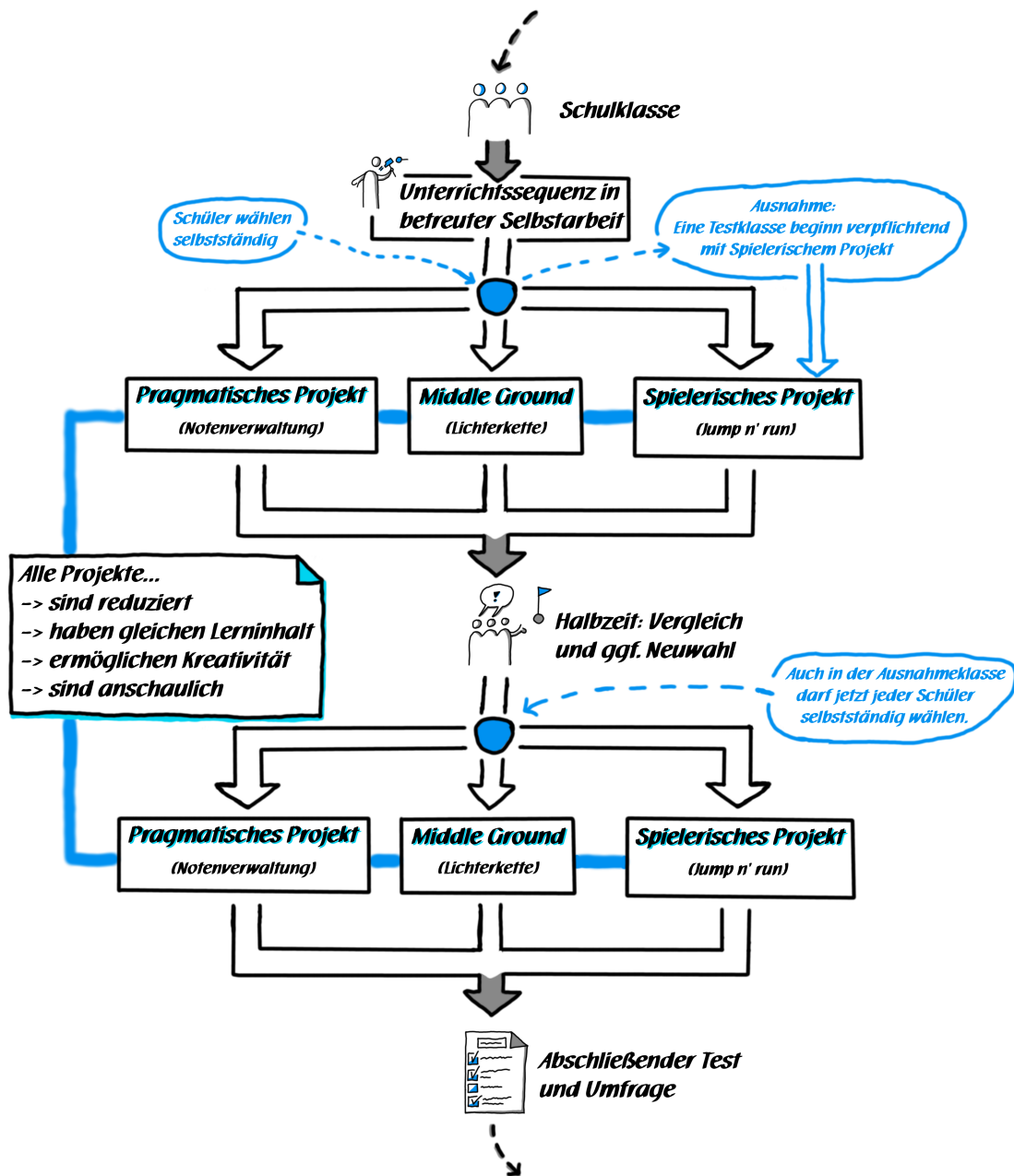


Figure 4.2: Visualisierung des Ablaufs der zweiten Testreihe

Aufteilung der Datensätze		
Testgruppe	Teilnehmende Klassen	Teilnehmende Schüler
Vorher und Nachher Vergleich	5	86 ¹
Unterricht ganz ohne Engine	3	54
Unterricht immer mit Engine	4	90
Insgesamt	11	230

Figure 4.3: Verteilung der Datensätze von Testlauf 1 auf die drei verschiedenen Erhebungsgruppen

5 Ergebnisse und Einsichten aus den Tests

Die Testläufe 1 und 2 sind an Schulen in Bayern ausgeführt worden. Hierbei wurden insgesamt *ca. 400 Datensätze* gewonnen. Aus dieser Informationsmenge lassen sich vielfältige Einsichten gewinnen. Daher seien im folgenden bemerkenswerte Zusammenhänge beschrieben zusammen mit einer belegten Interpretation.

Dabei werden die Einsichten aus den Testläufen isoliert betrachtet. Zusammenhänge zwischen den Ergebnissen der Tests lassen sich zwar auch interpretieren, jedoch sind solche Bestrebungen mit der Menge an Daten nicht ausreichend fundiert, um vertrauenswürdige Ergebnisse zu produzieren.

5.1 Einsichten aus Testlauf 1

Die Analyse konzentriert sich im Folgenden also auf die Ergebnisse aus den Testläufen für sich. Zunächst wird hierbei Testlauf 1 betrachtet.

5.1.1 Der Umstieg auf die Engine steigert Motivation und Selbstwirksamkeit

Eine Beobachtung, die bei der Datenanalyse schnell ins Auge fällt, sind die Veränderungen in den Erhebungen vor und nach dem Umstieg auf die Engine. Hierfür seien zunächst die primären Erhebungen zur Messung der Motivation in den Fokus gerückt:

- *Wahrgenommene Selbstwirksamkeit* (durch Umsetzung eigener Ideen)
- *Wahrgenommene Erfolgserlebnisse* (als Indikator für Motivation)
- *Vorfreude auf den Informatikunterricht* (als Indikator für Motivation)
- *Wahrgenommene Überforderung beim Bearbeiten von Aufgaben* (als Indikator für fehlende Selbstwirksamkeit)

Beim Vergleich dieser Parameter vor und nach dem Umstieg stellt man eine *deutliche Veränderung zum Positiven fest*: Alle genannten Parameter verbessern sich im Sinne der Motivation und Selbstwirksamkeit der Schüler deutlich (siehe Darstellung 5.1). Hier

können relative Veränderungen zwischen 15 und 40 Prozent zum Besseren gemessen werden. So zeichnet sich deutlich ab, dass die *Schüler nach dem Umstieg motivierter und selbstwirksamer am Unterricht teilnehmen*.

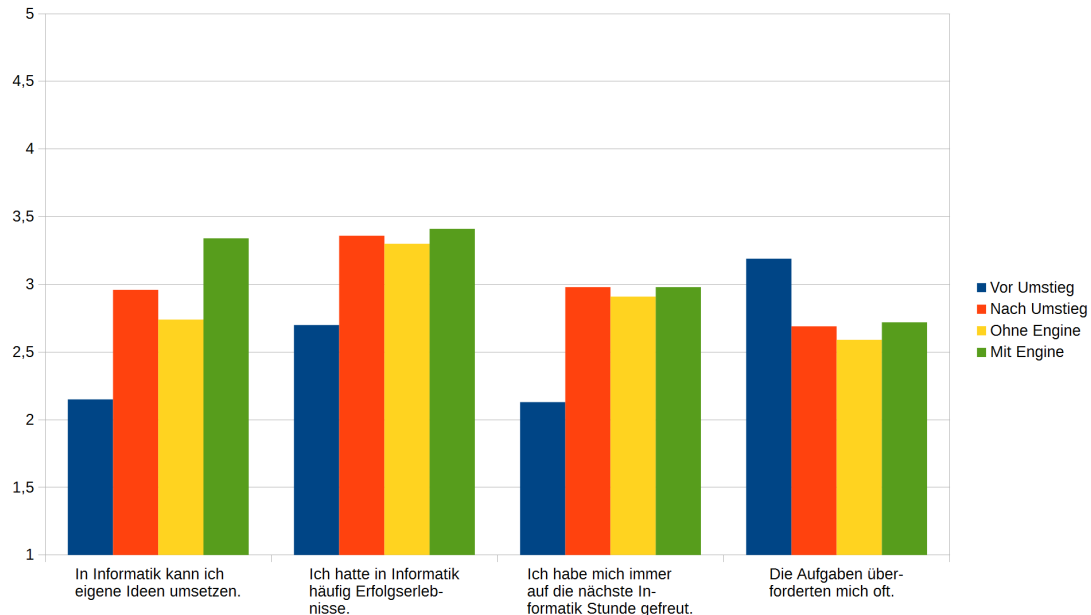


Figure 5.1: Vergleich der erhobenen Motivationsfaktoren nach Schülerwahrnehmung in Testlauf 1 (jeweils Durchschnitt, Antworten von 1 bis 5 in steigender Zustimmung möglich)

Dies spricht zunächst *deutlich für These 1*. Die Effekte des Umstiegs lassen sich als Ergebnis von mehr *Autonomie* (durch mehr Freiheit bei der Entwicklung im Framework) und *mehr Erfolgserlebnisse* (zu ermessen an 5.3, durch klarere, weniger überladene Inhalte). Diese Interpretation lässt sich durch Ergebnisse aus der Forschung untermauern. So ist der Zusammenhang zwischen Unterricht, der Schüler Autonomie ermöglicht, und intrinsischer Motivation durch mehrere Studien belegt (siehe [Ric00], S. 59); er hat auch in der populärwissenschaftlichen und in der Sachliteratur Einzug gefunden (vgl. u.A. [Pin11]).

5.1.2 Kritische Hinterfragung der vorigen Schlussfolgerung

Die eben genannte Erkenntnis ist eine ganz wesentliche Untermauerung von These 1. Dies ist ein wertvolles Ergebnis, jedoch sollte dies zunächst nicht so hingenommen werden, ohne dies auch aus einem *kritischen Standpunkt* zu betrachten. Nicht zuletzt,

um einem möglichen Bias entgegenzuwirken, sei im folgenden eine *Anfechtung* der These angestellt.

Für sich betrachtet zeigen die Ergebnisse im Vorher-Nachher-Vergleich einen deutlichen Anstieg an Motivation. Bezieht man nun noch die Werte der Vergleichsklassen (ebenfalls in Darstellung 5.1) ein, stellt man fest, dass die Bewertung von Schülern *ohne Kontakt mit der Engine* hier in den meisten Bereichen keine signifikanten Unterschiede aufweist zu den Bewertungen von Schülern, die *nach dem Umstieg auf die Engine* an der Evaluation teilgenommen haben. Dies kann unterschiedlich interpretiert werden. Zwei mögliche Szenarien liegen nahe:

1. Die Schulklassen, die am Vorher-Nachher-Programm teilgenommen haben, hatten *bisher prinzipiell qualitativ unterdurchschnittlichen* Unterricht, was in schlechterer Motivation zu Beginn niederschlägt. Nach dem Umstieg auf die Engine und die bereitgestellten, gut geplanten Unterrichtssequenzen kommen die Schüler auf das zu erwartende Niveau. Die Effekte der Engine wären somit sekundär zum Einfluss des vorangegangenen Unterrichts.
2. Die zu Beginn schlechteren Werte sollten in einer idealen Testumgebung nicht vorkommen. Die quantitativen Erhebungen *vor Umstieg und ohne Engine sollten ähnlich sein*, da hierbei von ausreichend äquivalenten Ausgangsbedingungen zu ausgehen ist. Daher ist der *Einfluss des Testablaufs* an sich mit verantwortlich für die Ergebnisse: Schüler könnten z.B. in der ersten Erhebung ihre Selbsteinschätzung eher gering ansetzen, in der Erwartung, dass der Umstieg ihre Fertigkeiten verbessert. Dies kann als Ausdruck des *Hawthorne-Effekt* auf die Studie betrachtet werden und ist bei derartigen Erhebungen nicht ungewöhnlich (siehe [Rya04] S. 23 f.).

Diese Interpretationen werden im Folgenden kurz diskutiert:

1. *Die Schüler hatten qualitativ unterdurchschnittlichen Unterricht*

Die Argumentation ist plausibel, jedoch sprechen wesentliche Aspekte dagegen:

- *Es gibt eine Vielzahl an teilnehmenden Klassen*

Die Evaluation stützt sich auf Daten von *über 10 Schulklassen* verschiedener Schulen und betreut von verschiedenen Lehrern. (siehe Darstellung 4.3) Die betrachteten Daten sind ein *Durchschnitt über den jeweiligen Testgruppen*; ein solcher Ausschlag ist also nicht unmittelbar plausibel.

- *Die Ergebnisse aus den Tests widersprechen der Vermutung*

Betrachtet man die Ergebnisse der *normierten Leistungserhebung* (siehe Darstellung 5.2), so erbringen hier die Schulklassen vor Umstieg vergleichbare (sogar

leicht bessere) Testergebnisse mit denen der Schulklassen, die ohne Engine unterrichtet wurden. Damit stehen die Schulklassen *auf Leistungsebene nicht zurückgeschlagen*, wie die Ergebnisse der Meinungsabfrage in Darstellung 5.1 vermuten ließen. Eine solche Form von "Positive Performance Feedback" erhöht normalerweise die intrinsische Motivation (siehe [Ric00], S. 59).

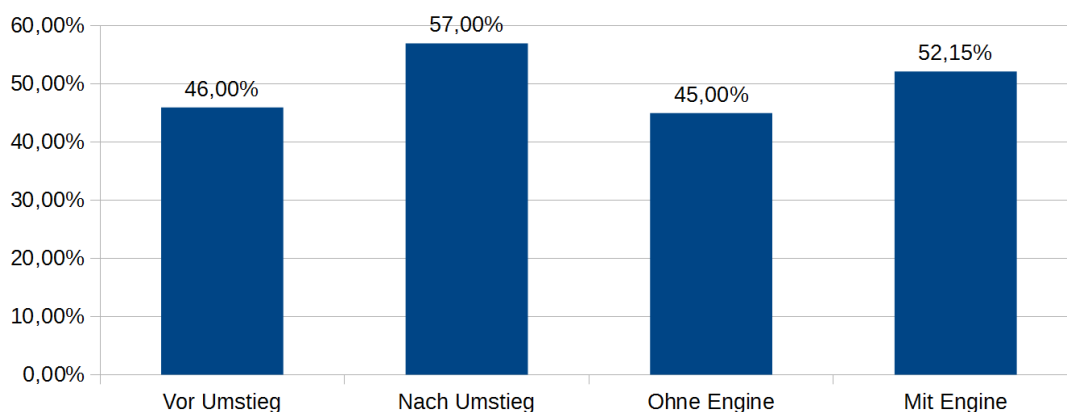


Figure 5.2: Ergebnisse der Leistungserhebungen in Testlauf 1 (jeweils Durchschnitt)

Damit lässt sich abschließend sagen, dass diese Interpretation wohl nicht ohne Weiteres zutrefflich ist.

2. Die Schüler waren durch die Testumgebung selbst beeinflusst

Der mögliche Einfluss der Testumgebung ist nicht zu bestreiten. Die Schüler wurden mit der nötigen Sorgfalt auf die Prozesse hingewiesen. Einflüsse wie der Hawthorne-Effekt lassen sich also *kaum vermeiden und nicht ausschließen*. Die Frage ist also nun vielmehr, wie stark der Einfluss des Hawthorne-Effektes ist. Hierzu seien folgende Argumente angeführt:

- *Alle Erhebungen unterlagen demselben Effekt*

Die Umgebung, in der die Daten erhoben wurden (vor und nach der Umstellung, mit und ohne Engine), hat sich nicht verändert. Es ist davon auszugehen, dass die Effekte der Testumgebung sich damit im Vor- und Nachhervergleich nicht wesentlich unterscheiden. Die *relative Veränderung im Vor- und Nachhervergleich* verliert unter diesen Bedingungen auch mit Hawthorne-Effekt nicht an Wert für die Argumentation: da hierbei nur die *Veränderung an Variablen* (in diesem Fall: Die Engine als Tool im Unterricht) relevant ist, lässt sich das Ergebnis trotz möglichem Hawthorne-Effekt nutzen.

- Die fachlichen Leistungserhebung kann als objektiverer Maßstab hinzugezogen werden

Meinungsabfragen (wie abgebildet in Darstellung 5.1) sind aufgrund ihres subjektiven Charakters prinzipiell besonders anfällig für verzerrende Effekte. Unter Hinzunahme einer objektiveren Messung können die Ergebnisse jedoch sicherer fundiert werden. Unter Einbezug der *Leistungserhebung* (siehe Darstellung 5.2), ist festzustellen, dass die subjektive Wahrnehmung der Schüler durch die *objektiven Testergebnisse* gestützt wird. Die Schüler berichten nach dem Umstieg von mehr Erfolgserlebnissen und weniger gefühlter Überforderung. Dies deckt sich mit den Testergebnissen: Die Prüfungsleistung hat sich nach Umstieg um über zehn Prozentpunkte verbessert. Diese Entwicklung stützt die Annahme, dass die Schülerwahrnehmung nicht unverhältnismäßig verzerrt ist.

Damit lässt sich darlegen, dass die Einflüsse von verzerrenden Einflüssen wie dem Hawthorne-Effekt vermutlich nicht die Ergebnisse maßgeblich verfälschen.

Im Folgenden wird daher auf diese Effekte nicht weiter eingegangen.

Zusammengefasst kann also davon ausgegangen werden, dass *die Qualität des vorangegangenen Unterrichts keinen enormen Einfluss auf die Messungen hatte* (oder gleichwertig war) und desweiteren davon, dass *der Hawthorne-Effekt die Ergebnisse nicht maßgeblich verfälscht*.

Die Ergebnisse aus Kapitel 5.1.1 bestehen die Anfechtung.

5.1.3 Der Unterricht mit der Engine verbessert die Selbsteinschätzung und das Fachverständnis der Schüler

5.1.4 Der Unterricht mit der Engine verbessert die wahrgenommene Unterrichtsqualität

Nach dem Umstieg scheinen die Schüler den Unterricht, den sie erhalten mehr wertzuschätzen. Bei den weiteren Aspekten der Schülerbefragungen zeichnet sich dies deutlich ab (Siehe Darstellung 5.3):

- Die *Eigenarbeit am PC* sei besser in den Unterricht eingebunden, Inhalten werden so besser verstanden.
- Schüler verlieren den "Roten Faden" nicht so schnell aus den Augen. Die *Zusammenhänge zwischen den Unterrichtsstunden* werden besser erfasst.

- Die konkreten Unterrichtsbeispiele (beim Umstieg durch eine Verlagerung auf Computerspiele gekennzeichnet) werden als *wesentlich ansprechender* wahrgenommen.
- Die Schüler geben an, die Lerninhalte nach dem Umstieg *tendenziell schneller verstanden* zu haben.

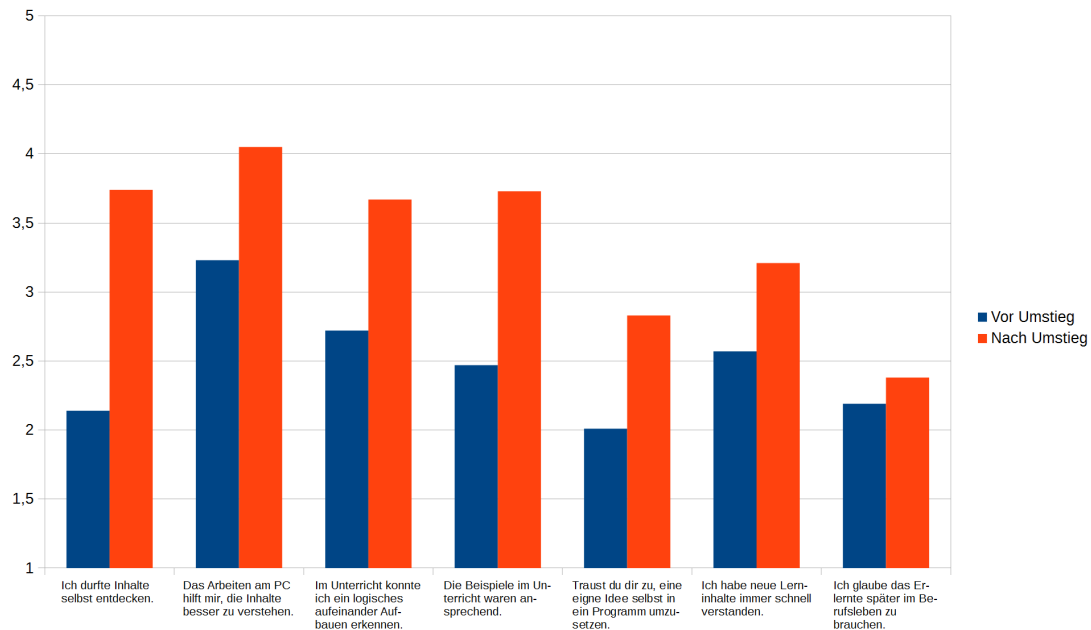


Figure 5.3: Vergleich der erhobenen sekundären Faktoren nach Wahrnehmung in Testlauf 1 (jeweils Durchschnitt, Antworten von 1 bis 5 in steigender Zustimmung möglich)

Diese Messungen decken sich mit der wahrgenommenen Vorfreude auf den Unterricht (siehe Darstellung 5.1).

5.2 Einsichten aus Testlauf 2

Das primäre Interesse an Testlauf 2 liegt bei der *Prüfungsleistung*, da hiermit die zweite These diskutiert werden kann. Bei diesem komplexeren Ablauf (siehe Darstellung 4.2) ermöglicht auch das *Wahlverhalten* der Schüler bei der Auswahl ihres Projektes potentiell interessante Einsichten ebenso wie die *Begründungen* hierzu.

5.2.1 Je spielerischer das Projekt, desto beliebter und motivierender ist es bei den Schülern

Geht es nach den Schülern, so lässt sich anhand der *Erst- und Neuwahl bei den freiwilligen Klassen* eine klare Beliebtheits-Skala für die drei Projekte aufbauen: (siehe 5.4, freiwillige Klassen)

1. Das *Spielerische Projekt* (mit 23 vor und 26 teilnehmenden Schülern nach Neuwahl)
2. Das *„Middle Ground“ Projekt* (mit insgesamt 12 teilnehmenden Schülern vor und nach Neuwahl)
3. Das *Pragmatische Projekt* (mit 3 Schülern vor Neuwahl, danach unbearbeitet)

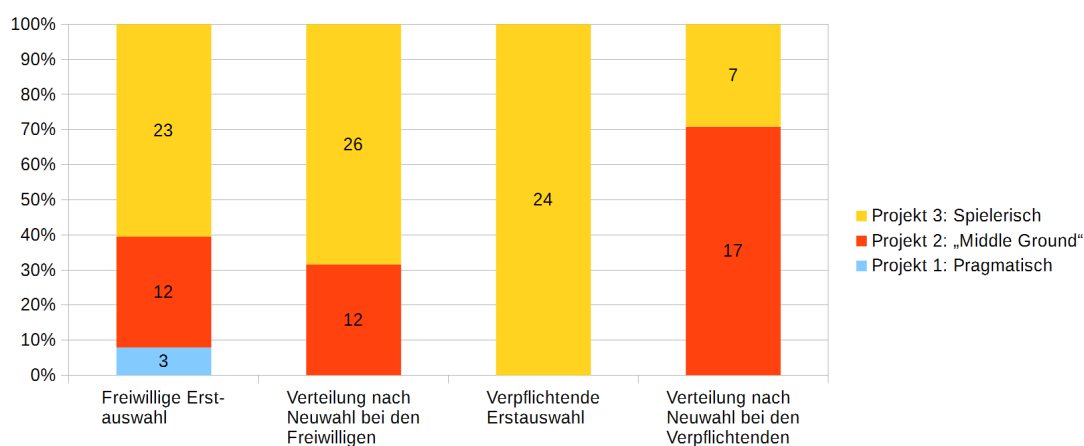


Figure 5.4: Übersicht über Projektauswahl der Schüler (absolute Schülerzahlen in den Balken, 0 wo keine Angaben)

Dieses Ergebnis lässt sich indirekt im Sinne der zweiten These interpretieren: Die *spielerische Komponente*, die die Schüler eher aus ihrer Lebenswelt abholt, kann die Schüler eher motivieren.

5.2.2 Eine verpflichtende Erstauswahl verändert den Bezug und die Motivation zum Unterricht

Aus der bereits angebrachten Darstellung 5.4 ist ersichtlich, wie unterschiedlich das Verhalten ausfällt zwischen den Klassen mit freiwilliger Erstwahl und den *Klassen mit verpflichtender Erstauswahl*. So fällt die Menge an Schülern, die das spielerische Projekt bearbeiten, nach der Neuwahl auf unter ein Viertel.

Offensichtlich ist also das Wechselverhalten hierdurch beeinflusst. In Anbetracht der *Gründe, die die Schüler für den Wechsel angeben*, (siehe Darstellung 5.5) werden die Ursachen für den Wechsel deutlich:

- *Die Schüler nehmen ihr verpflichtendes Projekt als anstrengender wahr*

So gibt ungefähr die Hälfte der Schüler an, dass ihr verpflichtendes Projekt ihnen zu anstrengend war. Von den Schülern, die von ihrem freiwillig gewählten Projekt wechselten, gibt dies *niemand* an. Dieser Kontrast ist bemerkenswert, da die Schüler an Projekten mit derselben Anforderungsstufe (siehe Kapitel ??) bearbeitet haben; mitunter das exakt selbe Projekt. Ein Schüler gibt sogar an, dass sein freiwillig gewähltes Projekt nicht fordernd genug war.

- *Die Schüler nehmen weniger Unterstützung durch die Lehrkraft wahr*

Unter den Gründen für den Wechsel findet sich nun auch die Aussage, die Lehrkraft habe nicht ausreichend Unterstützung geliefert für die Bearbeitung der Aufgaben. Auch dies ist bemerkenswert, da die Testläufe sich alle nicht in Lehrkraft und Material unterscheiden (siehe ebenfalls Kapitel ??).

Dieser Unterschied in der Anforderung des Projektes scheint eher subjektiv zu sein, betrachtet man die Testergebnisse der beiden Gruppen, die nicht bemerkenswert unterschiedlich ausfallen (siehe X und Y).

Somit scheint der Unterschied eher in der Wahrnehmung zu liegen, als daran

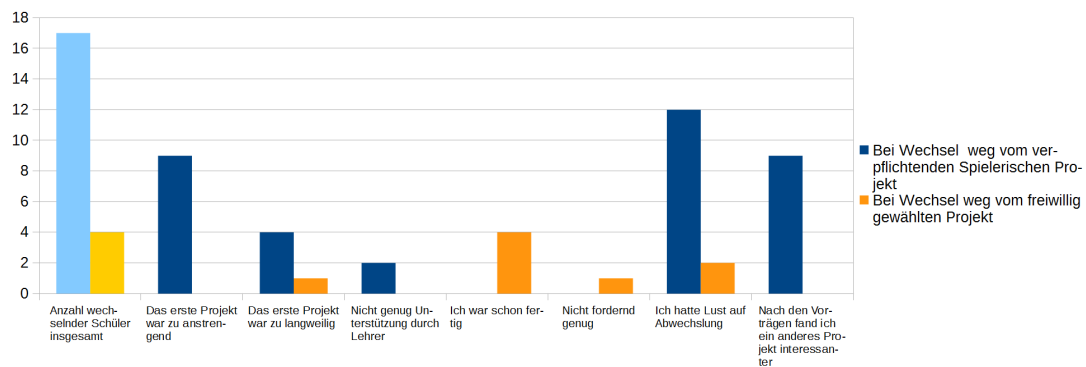


Figure 5.5: Angaben der Schüler zu ihrer Begründung des Projektwechsels

5.2.3 Wechselnde Schüler erzielen bessere Testergebnisse als bleibende

5.2.4 Wo vergleichbar,

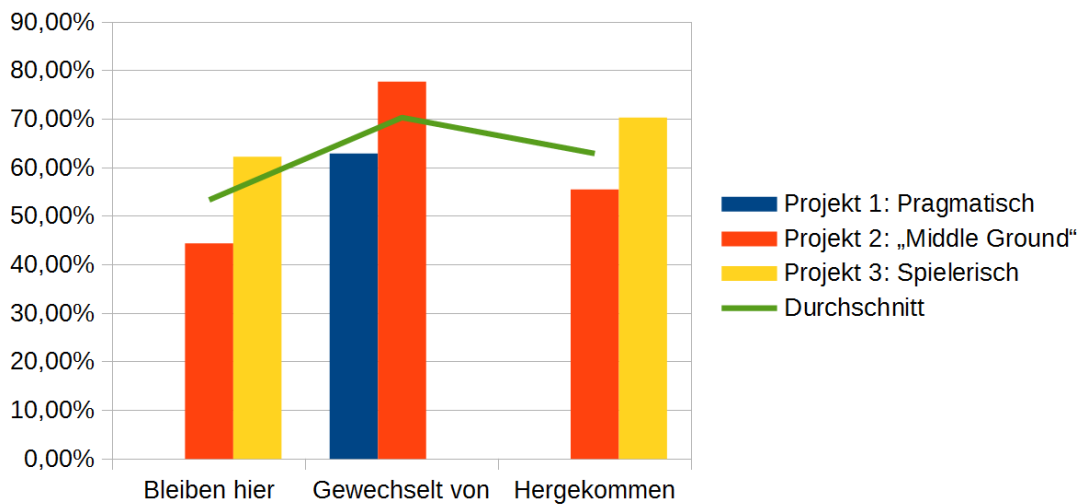


Figure 5.6: Ergebnisse der Leistungserhebung in Testlauf 2 (jeweils Durchschnitt, nach Wechselverhalten und Projektwahl). Balken fehlen überall, wo es keine Schüler gibt, die die Eigenschaften erfüllen.)

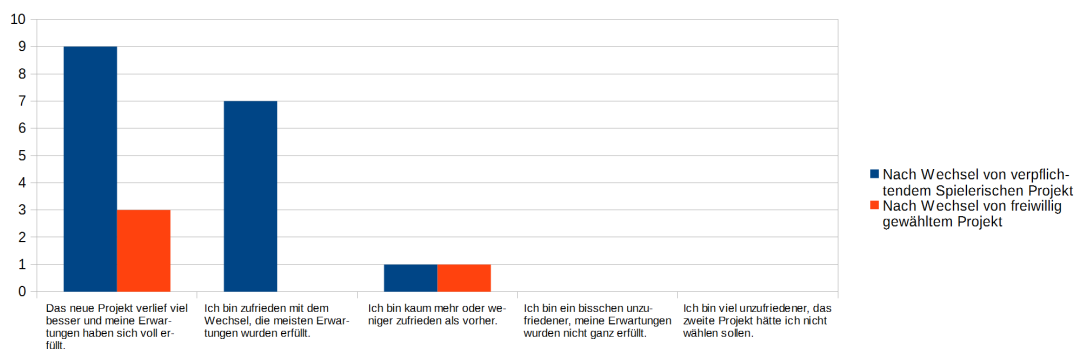


Figure 5.7: Angaben der Schüler zur Zufriedenheit mit ihrem Projektwechsel

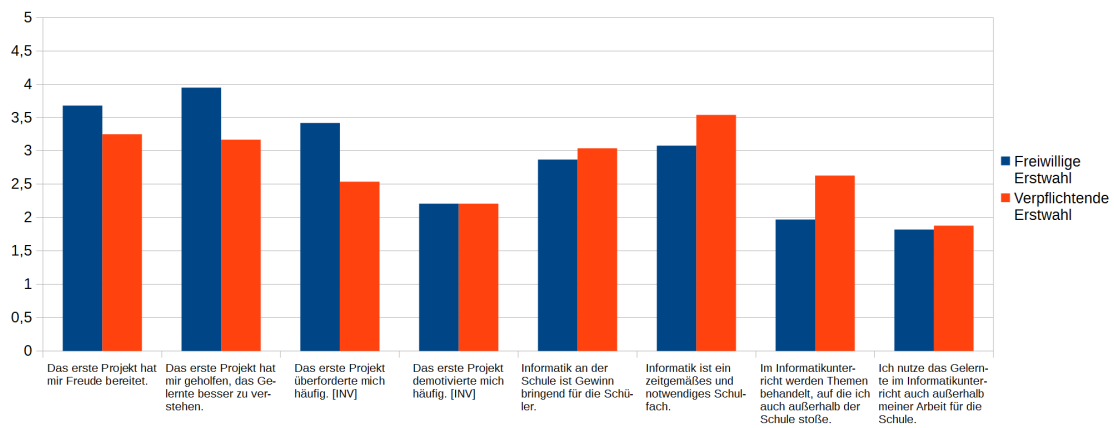


Figure 5.8: Bewertung der Sinnhaftigkeit des Unterrichts durch die Schüler, aufgeteilt in die freiwillige und verpflichtende Erstwahl (jeweils Durchschnitt)

6 Verifikation und Diskussion der Thesen

Mit den Erkenntnissen aus Kapitel 5 können nun These 1 und These 2 explizit diskutiert werden.

Nachdem der nötige Testlauf für These 3 nicht zum Zeitpunkt der Arbeit abgeschlossen werden konnte (siehe Kapitel 4), wird These 3 hier nicht berücksichtigt. Zu einem späteren Zeitpunkt lässt sich dies außerhalb dieser Arbeit nachholen. Näheres hierzu wird in Kapitel 7 erläutert.

6.1 Verifikation und Diskussion zu These 1

These 1

Bei Programmieraufgaben fördert *Simplizität und Rückhaltung* komplexer und (momentan) nichtrelevanter Lerninhalte in der Aufgabenstellung mehr *Kreativität* und damit *Motivation* bei den Schülern als ein *umfassendes Aufgabenspektrum*, das ganzheitliches Hintergrundverständnis entwickelt.

In Abschnitt 5.1.1 wurde bereits erläutert, dass die Ergebnisse aus Testlauf 1 im Sinne dieser These ausfallen: Nach dem Umstieg auf die Engine (die überdurchschnittliche Funktionalität unter starker Zurückhaltung bietet) im Unterricht, berichten Schüler von mehr Motivation, mehr Freiheiten und mehr Freude am Unterricht.

6.2 Verifikation und Diskussion zu These 2

These 2

Die *spielerische Komponente* bei Aufgaben führt zu mehr *Motivation und Lernzuwachs* im Informatikunterricht als die *pragmatisch anwendungsbezogene Komponente*.

Die Diskussion dieser These fällt komplexer aus.

6.3 Grenzen der Verifikation

7 Zukünftige Schritte im Zuge dieser Arbeit

Diese Arbeit stellt kein in sich abgeschlossenes Projekt dar. Wie bereits an verschiedenen Stellen angedeutet, gibt es verschiedene Arbeitsbereiche, die auch weit nach Veröffentlichung dieses Dokuments weitergeführt werden. Um diese Thesis abzurunden und einen Überblick über diese ausstehenden Aufgaben zu schaffen, seien die wichtigsten Arbeitsschritte im folgenden aufgeführt:

- *Weiterentwicklung der Engine Alpha Software*

Die Engine Alpha, auf deren Einfluss im Unterricht diese Arbeit aufbaut, ist im kontinuierlichen Prozess der Weiterentwicklung, es wird am Design und der Implementierung der *Version 4.0* gearbeitet (siehe branches in [ea_github]). Hierbei fließen nun auch Ergebnisse aus den Testläufen zurück, die die Gestaltung der neuen Funktionen und der Schnittstelle für die Schüler beeinflussen.

- *Erweiterung des Datenschatzes für die Testläufe*

Die Daten aus den Testläufen sind hilfreich und ermöglichen eine Vielzahl von Erkenntnissen. Jedoch sind diese in Kapitel 5 beschriebenen Ergebnisse durch die zeitliche und quantitative Beschränkung von *ca. 400 Datensätzen innerhalb eines Schuljahres* nur bedingt gültig. Um die Ergebnisse zu erweitern, zu untermauern und sie herauszufordern werden *weitere Evaluationen* angestrebt. So kann die *Unterrichtsgestaltung und Entwicklung mit der Engine* über die reine Qualität der Implementierung hinaus verbessert werden.

Hierfür sind Wiederholungen der Testläufe sowie die Umsetzung neuer Erhebungen geplant.

- *Umsetzung von Testlauf 3*

Bei den für die Zukunft angesetzten Erhebungen ist besonders die Organisation von Testlauf 3 zu erwähnen. Dieser war ursprünglich als Teil dieser Arbeit geplant, musste aber aufgrund von Anmeldefristen zu derartigen Schülerumfragen in das Schuljahr 2016/17 verschoben werden. Um die Arbeit in ihrem ursprünglich geplanten Datenumfang zu realisieren, wird dieser Testlauf auch nach der Veröffentlichung angestrebt.

8 Abschließende Bewertung des Lehrplans im Zuge der Ergebnisse

9 Abschließende Worte

Change is the end result of all true learning.

Leo Buscaglia

List of Figures

4.1	Ablauf von Testlauf 1	10
4.2	Ablauf von Testlauf 2	13
4.3	Testlauf 1: Verteilung der Datensätze	14
5.1	Testlauf 1: Primäre Motivationsfaktoren	16
5.2	Testlauf 1: Ergebnisse in der Leistungserhebung	18
5.3	Testlauf 1: Sekundäre Motivationsfaktoren	20
5.4	Testlauf 2: Verteilung bei der Projektauswahl	21
5.5	Testlauf 2: Gründe für den Projektwechsel	22
5.6	Testlauf 2: Ergebnisse der Leistungserhebung (Haupt-Klassen)	23
5.7	Testlauf 2: Zufriedenheit nach Projektwechsel	23
5.8	Testlauf 2: Unterrichtsbewertung durch die Schüler	24

List of Tables

Bibliography

- [BLV] BLVV Bayern. *Zeitplan für den LehrplanPLUS*. <https://www.bllv.de/Zeitplan-Einfuehrung-Lehrplan-GS.9748.0.html>. Zugriff: 14.12.2016.
- [Elk08] W. A. Elke Frey Christian Heidrich. *Informatik am Naturwissenschaftlich-technologischen Gymnasium, Jahrgangsstufe 10*. Staatsinstitut für Schulqualität und Bildungsforschung, 2008.
- [Enga] Engine Alpha Website. *API Dokumentation (v3.1.1)*. <https://docs.engine-alpha.org/v3.1.1/>. Zugriff: 14.12.2016.
- [Engb] Engine Alpha Website. *Lehrerbereich*. <http://engine-alpha.org/wiki/Lehrer>. Zugriff: 14.12.2016.
- [Engc] Engine Alpha Website. *Tutorial-Übersicht*. <http://engine-alpha.org/wiki/Tutorials>. Zugriff: 14.12.2016.
- [Gan16] M. Ganshorn. *Handreichung: Lehrplanbewertung und -gegenüberstellung für das Studienseminar Informatik des Adlabert-Strifter-Gymnasiums, Passau*. Im digitalen Anhang der Arbeit. 2016.
- [Kula] Kultusministerium Bayern. *Kompetenzorientierung und LehrplanPLUS*. <http://www.isb.bayern.de/schulartuebergreifendes/paedagogik-didaktik-methodik/kompetenzorientierung>. Zugriff: 14.12.2016.
- [Kulb] Kultusministerium Bayern. *Lehrplan 10. Klasse Informatik am bayerischen Gymnasium (NTG 2)*. <http://www.isb-gym8-lehrplan.de/contentserv/3.1.neu/g8.de/index.php?StoryID=26435>. Zugriff: 14.12.2016.
- [Kulc] Kultusministerium Bayern. *Vorbemerkungen zum Lehrplan*. <https://www.isb.bayern.de/gymnasium/lehrplan/gymnasium/einleitendes/1023/>. Zugriff: 14.12.2016.
- [Pin11] D. Pink. *Drive: The Surprising Truth About What Motivates Us*. Riverhead Books, 2011.
- [Ric00] E. L. D. Richard M. Ryan. "Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions." In: *Contemporary Educational Psychology* 25.1 (2000), pp. 54–67. DOI: <http://www.sciencedirect.com/science/article/pii/S0361476X99910202>.

- [Rya04] W. A. Ryan Olson Jessica Verley. "What we teach students about the Hawthorne studies: A review of content within." In: *The Industrial-Organizational Psychologist* 41.1 (2004), pp. 23–39. DOI: http://www.siop.org/tip/backissues/Jan%2004/pdf/413_023to039.pdf.
- [Staa] Staatsinstitut für Schulqualität und Bildungsforschung (ISB). *Lehrplan 10. Klasse Informatik am bayerischen Gymnasium (LehrplanPlus)*. <http://www.lehrplanplus.bayern.de/fachlehrplan/gymnasium/10/informatik>. Zugriff: 14.12.2016.
- [Stab] Staatsinstitut für Schulqualität und Bildungsforschung (ISB). *Lehrplan 11. und 12. Jahrgangsstufe - Informatik*. <http://www.isb-gym8-lehrplan.de/contentserv/3.1.neu/g8.de/index.php?StoryID=26193>. Zugriff: 14.12.2016.
- [Stac] Staatsinstitut für Schulqualität und Bildungsforschung (ISB). *LehrplanPLUS - Gymnasium - Informatik - Fachlehrpläne*. http://www.lehrplanplus.bayern.de/schulart/gymnasium/fach/informatik/inhalt/fachlehrplaene?w_schulart=gymnasium&wt_1=schulart&w_fach=informatik&wt_2=fach. Zugriff: 14.12.2016.