
Softwareprojekte im Informatikunterricht
Projektarbeit – „Ganz schön clever“

Malte Voß, 16. Juni 2020

1 Vorwort

In der modernen Berufswelt gehört die Projektarbeit zur Normalität. Dabei werden gewünschte Produkte spezifiziert und umgesetzt. Zur Durchführung von Projekten gibt es Konventionen und Modelle, welche die Realisierung unterstützen.

Bereits im Informatikunterricht können Schülerinnen und Schüler Projekte umsetzen und so Methoden und Anforderungen aus der Berufswelt kennen lernen. Darüber hinaus sind weitere Inhalte der Informatik Teil des Projekts je nach thematischer Umsetzung.

Durch die Projektarbeit steht das Handeln der Schülerinnen und Schüler im Vordergrund. Prozessbezogene Kompetenzen werden zur Geltung gebracht. Insbesondere die Kooperation im Team ist eine wichtige Voraussetzung für den Erfolg des Projekts.

2 Verankerung im Bildungsplan

Der Bildungsplan 2016 für das Fach Informatik ¹ stellt den Rahmen für die Umsetzung des Projekts im Unterricht.

2.1 Inhaltsbezogene Kompetenzen

An inhaltsbezogenen Kompetenzen stehen folgende Gesichtspunkte im Vordergrund:

- Aspekte des Projektmanagements und ein Vorgehensmodell erklären.
Diese Kompetenz ist im Rahmen eines Softwareprojekts gut vermittelbar, da Begriffe wie Milestone, User Story, Sprint, etc. auch erlebbar sind und ihre Bedeutung damit greifbarer ist.
- Entwicklung eines Prototypen mit Methoden eines Vorgehensmodells.
Mit der Entwicklung eines Prototypen wird ein konkretes Ziel für das Projekt gesteckt. Die Verwendung der Werkzeuge eines Vorgehensmodells dient dabei als Unterstützung zur Umsetzung des Projekts in Teamarbeit.
- Anwendung erlernter Praktiken wie objektorientierte Programmierung.
Im Projekt werden bereits erlernte Kompetenzen benötigt, etwa beim Entwurf mit Hilfe des objektorientierten Paradigmas.

¹<http://www.bildungsplaene-bw.de/,Lde/LS/BP2016BW/ALLG/GYM/INF>

3 Projekt „Ganz schön clever“

Ganz schön clever ist ein Würfelspiel von Schmidt, es existiert auch eine online spielbare Version ² für eine Person. Es besteht eine gewisse Ähnlichkeit zu Kniffel, da durch den Wurf mit mehreren Würfeln Aufgaben auf einem Zettel gelöst werden.

4 Didaktische Analyse

4.1 Vernetzung des Bildungsplaninhalte

Im Softwareprojekt wird auf verschiedene inhaltliche Kompetenzen zurückgegriffen. Aufbauend auf bereits erlernte Kompetenzen eignet sich das objektorientierte Programmierparadigma besonders zum Entwurf der Software. Inkarniert durch reale Objekte ergeben sich schnell Klassen für Spieler, Würfel, Block, Felder und Aufgaben. Konzepte wie Geheimnisprinzip, abstrakte Klassen und Vererbung (Aufgaben und Felder sind polymorph) sind eingängig anwendbar. Zur Haltung von mehrer gleicher Datentypen bedarf es Datenstrukturen, möglich ist eine Realisierung mit Arrays, um etwa einzelne Felder in einer Aufgabe zu halten. Zur Berechnung der Punktzahl müssen die einzelnen Aufgaben und Felder traversiert werden. Wie in der Berufswelt werden weitere Werkzeuge zur Implementierung herangezogen. Versionierungssoftware wie git ermöglicht asynchrone Teamarbeit und die Nutzung bestehender Software wie GUI-Frameworks ist grundlegend für die Umsetzung von Spielen.

4.2 Didaktische Reduktion

4.3 Methodische Überlegungen

Zur Umsetzung des Softwareprojekts ist Scrum als Vorgehensmodell gewählt. Für Schülerinnen und Schülern als Anfänger in der Softwareentwicklung eignet sich dieses agile Vorgehen besser als traditionelle Prozessmodelle, da bereits die Festlegung der Anforderungen in Form von User Stories ein geringeres Abstraktionsniveau besitzt. Weiterhin ist durch das zyklische Modell eine Anpassung der Ziele mit jedem neuen Sprint möglich. Damit bietet Scrum den Entwicklern eine flexiblere Planung und verlangt weniger Einschätzung im Voraus.

GUI-basierte Software bietet einfache Möglichkeiten zum Testen. Dabei gehen für Spiele Anforderungen und Testfälle meist Hand in Hand, da sie von bestimmten Szenarios und Eingaben ausgehen. Die Überprüfung dieser Testfälle geschieht

²<https://www.schmidtspiele.de/static/onlinespiele/ganz-schoen-clever/>

dann analog. Testmethoden wie jUnit eignen sich nur in geringem Umfang, da die Erzeugung von Szenarios aufwendig ist.

Das Softwareprojekt eignet sich auch zur Umsetzung im Flipped-Classroom-Konzept. So können Inhalte bereits vorab per Video verfügbar gemacht werden. Im Unterricht kann die Lehrkraft diese Inhalte aufgreifen oder bei Schwierigkeiten helfen. Mit dem Flipped-Classroom-Konzept kann die Lehrkraft die Entwicklung des Projektes besser verfolgen, da die Unterrichtszeit durch das Handeln der Schülerinnen und Schüler genutzt wird.

4.4 Unterrichtsziele

Der Unterricht ist stark durch das Produkt am Ende des Projekts motiviert. Dabei bietet die Umsetzung des Spiels große Freiheiten, schon beginnend bei der Erstellung der Anforderungen an die Anwendung. Man sieht, dass es viele Wege zum Ziel gibt. Dabei ist die Festlegung des Weges ein Teil der Projektplanung

5 Umsetzung des Projekts

5.1 Projektauftrag

Die Anforderungen an das Projekt sind in Form von User Stories festzuhalten. Beispielhaft sind hier aufsteigend grob priorisiert ein paar gegeben:

- Man kann die Würfel würfeln.
- Ein ausgewählter Würfel lässt sich nur in ein legales Feld eintragen.
 - aufteilbar in verschiedene Farben.
- Der eingetragene Würfel wird bei Seite gelegt.
- Ausgeschiedene Würfel (Augenzahl kleiner als eingetragener Würfel) werden aufs Silbertablett gelegt.
- Für jeden Spieler werden die Punkte gezählt.
- Joker können erspielt werden.
- Aktive Joker können verwendet werden.
- ...

Diese Sammlung an User Stories ist der Product Backlog.

5.2 Projektphasen

Zu Beginn des Softwareprojekts im Informatikunterricht lernen die Schülerinnen und Schüler das Spiel selbst kennen. Daraufhin kann der Product Backlog in den einzelnen Teams erstellt werden.

In den folgenden Phasen werden Sprints durchlaufen, das heißt, einzelne User Stories werden aus dem Product Backlog ausgewählt und implementiert. Die Planung der Sprints obliegt den Schülerinnen und Schülern, die Lehrkraft kann jedoch unterstützen und beraten, beispielsweise durch Einschätzung des Aufwands.

Beispielsweise lassen sich folgende Sprints mit Zielen definieren:

1. Fachklassen für Würfel, Spieler, Blatt, Aufgabe, Feld implementieren
2. GUI für die obigen Klassen erzeugen
3. Controller implementieren
4. Update-Methoden für die GUI hinzufügen
5. weitere Regeln wie Runden, Joker implementieren
6. Abschlusspräsentation

5.3 Beschreibung des eigenen Prototypen

Ein möglicher Lösungsansatz ist auf GitHub verfügbar ³. Hier wird JavaFX zur graphischen Umsetzung verwendet. Das Layout der Benutzeroberfläche ist dabei großteils in separaten Dateien (`.fxml`) festgehalten. Mit JavaFX können komplexe Strukturen der GUI-Programmierung wie Listener versteckt und durch einfach Funktionsaufrufe realisiert werden. Im Lösungsvorschlag gibt es eine Oberklasse für graphische Objekte, die das Laden der `.fxml`-Datei zur Verfügung stellt. Dabei wird die graphische Repräsentation mit dem Objekt verknüpft, welches als Controller für die graphischen Elemente dient. Dadurch wird Darstellung und Modell zwar nicht getrennt, jedoch sind die Zusammenhänge dazwischen einfacher zu finden.

Im Modell gibt es mehrere Spieler, die jeweils ein Blatt haben. Ein Blatt hat 5 Aufgaben, die jeweils wiederum aus Feldern bestehen. Die Felder sind je nach Aufgabe entweder ankreuzbar oder eintragbar. Weiterhin gibt es genau sechs Würfel, die in einem `enum` vorhanden sind. Die Verwendung eines `enum` bringt den Vorteil, dass Instanzen Singletons sind.

³<https://github.com/vossmalte/GSC>

6 Unterrichtliche Realisierung

Das Projekt ist für Gruppen von drei bis vier Personen vorgesehen. Der Zeitumfang beträgt im Basisfach etwa 6 Wochen.

Ein Softwareprojekt zu benoten ist komplex, denn es wird eine Leistung über einen langen Zeitraum, an der mehrere Schülerinnen und Schüler beteiligt sind, bewertet. Wie in vorherigen Teilen des Projekts bietet sich auch hier die Möglichkeit, die Schülerinnen und Schüler mitwirken zu lassen und so transparente Kriterien zu erstellen. Diese Festlegung sollte bereits zu Beginn des Projektes geschehen und ist als Pendant zu Abnahmekriterien zu verstehen. Beispielfhaft kann ein Bewertungsschema so aussehen:

30% Spiel (Regeln, Hilfestellung, Spaß)

25% Entwurf

15% Dokumentation des Quellcodes

15% Design der GUI

15% Dokumentation des Softwareprojekts (Taskboards, Sprint backlogs, Testprotokolle)