Name: _____ Klasse: _____ Datum: _____

Projekt: Mein eigenes Spiel

Ziel: Du entwickelst ein eigenes Spiel für den PicoBoy – von der Idee über den Plan bis zum fertigen Programm. Du baust es in Stufen, testest es sorgfältig und stellst es am Ende der Klasse vor.

So läuft das Projekt ab

Schritt	Was tust du?	Seite
1. Idee und Plan	Spielidee finden, Planungsbogen ausfüllen, mit der Lehrkraft besprechen	1–2
2. Bauen und testen	in Stufen bauen, nach jeder Stufe testen und ins Logbuch schreiben	3
3. Vorstellen	Spiel der Klasse zeigen, Rückmeldung geben und bekommen	4

Aufgabe 1 ⭐ Eine Idee finden

Wähle eine Idee aus der Ideenbörse – oder erfinde eine eigene. Wichtig: Das Spiel soll **klein** sein. Ein kleines Spiel, das läuft, ist besser als ein großes, das nicht fertig wird.

Idee	Darum geht es	Wichtige Blöcke
Fang den Apfel	Ein Korb fängt fallende Äpfel.	Sprites, berührt Sprite, Zufallszahl
Meteoritenregen	Ein Raumschiff weicht Meteoriten aus.	Sprites, berührt Sprite, LEDs
Lava-Labyrinth	Eine Figur läuft durch ein gemaltes Labyrinth ins Ziel.	Canvas, berührt Farbe
Reaktionstest	Wer drückt am schnellsten, wenn die LED angeht?	Zeit, Zufallszahl, Tasten
Simon sagt	Eine Farbfolge merken und nachspielen	Listen, Töne, LEDs
Kipp-Kugel	Eine Kugel durch Kippen des Geräts ins Loch rollen	Beschleunigung, Sprites

Meine Idee: _____

Das gehört in jedes Spiel

- mindestens **eine Figur**, die man steuern kann
- mindestens **zwei Regeln** („Wenn …, dann …“), z. B. Punkte oder Treffer
- eine **Schleife**, in der die Regeln immer wieder geprüft werden
- ein **Ende**: gewonnen, verloren oder ein Punktestand
- mindestens **zwei eigene Funktionen** mit sprechenden Namen

Aufgabe 2 ⭐ Der Planungsbogen

Name des Spiels	
Worum geht es? 1–2 Sätze	
Figuren (Sprites) Name und Bild	
Steuerung Welche Taste macht was?	
Wie endet das Spiel? gewonnen / verloren	
Variablen Name und Startwert	

Die Regeln: Wenn ..., dann... (mindestens 4)

Wenn ...	dann ...

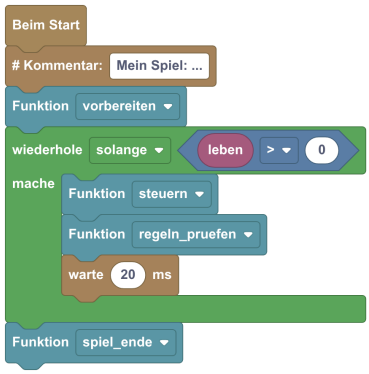
Skizze des Bildschirms

Funktionen sprechende Namen ohne Umlaute	
Muss ohne das geht es nicht	
Kann wenn noch Zeit ist	

Tipp: Ein ausgefülltes Beispiel findest du im Buch, Kapitel 21.1.

☐ Plan mit der Lehrkraft besprochen Unterschrift: _____

Aufgabe 3 ⭐ Bauen in Stufen – mit Bau-Logbuch



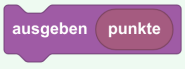
Beginne mit dem Hauptprogramm und leeren Funktionen – so wie links. (Deine Lehrkraft kann dir dieses Gerüst als Datei geben.) Baue dann **eine Stufe nach der anderen**: zuerst die Steuerung, dann die erste Regel, dann die nächste …

Nach **jeder** Stufe: testen, ins Logbuch schreiben, **speichern** – am besten als neue Datei (`spiel_stufe1.pbcp`, `spiel_stufe2.pbcp` …).

Datum	Was habe ich gebaut?	Was soll passieren?	Was passiert?	ok?

Ein Fehler, den ich gefunden habe – und wie ich ihn behoben habe:

Wenn etwas nicht klappt: ☐ Hängt der Block wirklich im Programm (unter „Beim Start“ oder in einer Funktion)? ☐ Steht der „falls“-Block in der Schleife? ☐ Wird die Funktion irgendwo aufgerufen? ☐ Steht „warte 20 ms“ in der Schleife? ☐ Meldung lesen – Erklärungen im Buch, Anhang B. ☐ Werte im Terminal anzeigen lassen:



Aufgabe 4 ⭐⭐ Testen lassen

Gib dein Spiel jemandem aus der Klasse. Sag nichts – schau nur zu. Was hat die Person anders gemacht, als du dachtest? Was hast du danach verbessert?

Aufgabe 5 ⭐ Vorstellen (etwa 3 Minuten)

1. Sag, wie dein Spiel heißt und worum es geht.
2. Zeig es im Emulator oder auf dem PicoBoy.
3. Zeig dein Hauptprogramm und **eine** Funktion, auf die du stolz bist.
4. Erzähl von einem Fehler, den du gefunden hast.

Aufgabe 6 ⭐ Rückmeldung: Zwei Sterne und ein Wunsch

Für das Spiel von: _____

⭐	
⭐	
Wunsch	

Aufgabe 7 ⭐⭐⭐ Version 2

Baue etwas aus deiner **Kann**-Liste ein oder setze den Wunsch aus der Rückmeldung um. Vergiss das Logbuch nicht.

Rückblick: Was ist an deinem Spiel richtig gut geworden? Was würdest du beim nächsten Mal anders machen?

Das habe ich gelernt:

☐ Ich kann ein Spiel mit einem Planungsbogen planen.

☐ Ich kann ein Programm in Funktionen aufteilen.

☐ Ich baue in Stufen und teste nach jeder Stufe.

☐ Ich kann einen Fehler systematisch suchen und beheben.

Mehr dazu im Buch: Kapitel 21 – Ein eigenes Spiel entwickeln · Anhang B – Fehlermeldungen verstehen