

# Gruppen-Projekt: Spielimplementierung

## Grundsätzliche Aufgabenstellung

### Aufgabe 1 – Gruppen- und Spielwahl

Findet euch in Kleingruppen von maximal drei Teilnehmern zusammen. Und entscheidet euch für eines der Spiele (ohne GUI):

- TicTacToe (\*-\*\*)
- Vier Gewinnt (\*\*)
- Schach: Eigene Version (\*\*\*)

### Aufgabe 2 – Git

Macht euch einen neuen Branch von main mit dem Namen des Spiels und euren Nachnamen.

**Beispiel:** von main abgehend mit *PongMusterfrauMustermann*

### Aufgabe 3 – Ordner/Packages

Legt euch unter Abgabe – src ein Package an mit dem Namen *SpielnameNachnameNachnameNachname*.

**Beispiel:** *PongMusterfrauMustermann*

Wie ihr das Package organisiert, ist euch überlassen.

Alle Dokumente etc., die ihr für das Projekt anlegt, müssen am Ende in eurem Package zu finden sein!

### Aufgabe 4 – Spielen und Szenarien

Macht euch Gedanken über die verschiedenen Spielszenarien. Spielt dazu das Spiel und notiert euch in einer Datei, welche Szenarien ihr in eure Implementierung des Spiels übernehmen möchtet.

Erstellt ein **Storyboard**. (Beispiel im Anhang & Handschrift ist erlaubt!)

Fr  Aufgabe 5 – Planung

Füllt die anliegende Planungsmatrix (Wochenplanung) aus, um euch tägliche Ziele zu formulieren und euren Arbeitsablauf zu strukturieren. (Anpassungen bitte markieren.)

Mo  Aufgabe 6 – Planung der Umsetzung

Plant die Umsetzung eures Spiels und legt dafür passende Klassen und Methoden an (diese müssen noch NICHT implementiert sein!!).

Es hilft auch ein Klassendiagramm zu entwerfen (keine Pflicht).

**Hinweis:** Ihr könnt Tipps als Hilfestellung bekommen.

## Gruppen-Projekt: Spielimplementierung

### Di Aufgabe 7 – Implementierung

Implementiert eure Version des Spiels.

Versucht dabei immer in kurzen Abständen zu commiten, damit ihr keine Konflikte auf dem Branch erhaltet.

### Mi Aufgabe 8 – Präsentation

Bereitet eine 15-minütige Präsentation zu eurem Entwicklungsprozess und euerem fertigen Produkt vor. Achtet dabei auf die allgemeinen Regeln zur Erstellung von Präsentationen. Wichtig ist, dass jeder ca. fünf Minuten Redeanteil erhält.

Nehmt gerne auch Probleme, Dinge, die gutgeklappt haben, etc. mit in die Präsentation. Und zeigt euren Entwicklungsprozess. (Ist wichtiger, als dass das Spiel fertig ist!)

Möchtet ihr vorab Feedback zu eurer Präsentation, dann müsstet sie bis 29.01.2025 um 12 Uhr bei Anita oder Alexandra abgegeben werden.

**Abgabe des Projektes und der Präsentation ist am 30.01.2025 um 12 Uhr**

**Wichtig:** Bei diesem Projekt wird es **KEINE Verlängerung** geben! Nehmt euch daher nur so viel vor, wie ihr in der Zeit schaffen könnt!

Ihr präsentiert am 30.01.2025 nachmittags im Tutorium.

#### Benotung:

- Es wird am Freitag und am Dienstag keine Pflichtaufgabe im klassischen Sinne geben. Dafür bekommt ihr auf das Projekt und die Präsentation eine Bewertung.
- Es muss erkennbar sein, dass jeder Teilnehmer Commits gemacht hat und sich am Projekt gleichermaßen, wie seine anderen Gruppenmitglieder, beteiligt hat!
- Wenn KI-Verwendung nicht kenntlich gemacht wurde, bekommen alle Teilnehmer ein rot!
- Es ist nicht schlimm, wenn das Spiel nicht fertig wird! Umso wichtiger ist es aber dann, euer Vorgehen und eure Arbeitsweise transparent zu machen und zu reflektieren.

**Ziel des Projektes** ist es, dass ihr lernt mit anderen zusammen zu arbeiten. Softwareentwicklung ist immer Teamarbeit! 😊

## Gruppen-Projekt: Spielimplementierung

### Bewertungskriterien

Kriterium	Inhalt	Punkte	Kommentar
<b>Planung</b>	Ihr habt einen Branch bei Git nach dem vorgegebenen Schema erstellt.	<b>4</b>	
	Ihr habt ein passendes Package erstellt.	<b>4</b>	
	In eurem Package ist eine sinnvolle Struktur erkennbar.	<b>4</b>	
	Alle notwendigen und verpflichteten Dateien sind in eurem Package auf Git zu finden.	<b>4</b>	
	Es gibt eine Datei, die erkennbar werden lässt, welche Spielszenarien ihr umsetzen möchten.	<b>6</b>	
	Ihr habt ein Storyboard angefertigt.	<b>5</b>	
	Ihr habt sinnvolle Ziele in euer Planungsmatrix festgehalten.	<b>5</b>	
	Veränderungen in der Planung wurden kenntlich gemacht.	<b>2</b>	
	Es ist ersichtlich, dass ihr euch gemeinsam Gedanken über die Planung der Implementierung gemacht habt. (Die Form ist nicht entscheidend.)	<b>6</b>	
<b>Umsetzung</b>	Ihr habt sinnvoll die OOP-Konzepte verwendet.	<b>6</b>	
	Der Code ist sauber, nutzt die gültigen allgemeinen Konventionen und beinhaltet keinen ungenutzten Code.	<b>7</b>	
	Es wurde ausreichend kommentiert.	<b>5</b>	
	Die Commit-Messages sind passend gewählt.	<b>5</b>	
	Wenn KI verwendet wurde, dann wurde dies vermerkt und der passende Prompt angegeben.	<b>2</b>	
	Jeder Teilnehmer hat zu gleichen Anteilen gearbeitet.	<b>5</b>	
<b>Präsentation</b>	Die Präsentation umfasst 15 Minuten.	<b>2</b>	
	Jeder Teilnehmer hat einen Redanteil von fünf Minuten.	<b>4</b>	
	Es wird der Entwicklungsprozess, die Planung und das Ergebnis präsentiert.	<b>6</b>	
	Es wird auf Probleme eingegangen.	<b>4</b>	
	Es werden positive Erlebnisse hervorgehoben.	<b>4</b>	
	Die PPP orientiert sich an den Anforderungen für Präsentationen, die wir auch zu den Exceptions aufgestellt haben.	<b>10</b>	
<b>Gesamt</b>		<b>100</b>	

## Gruppen-Projekt: Spielimplementierung

**Unsere Wochenplanung** (Tragt bitte eure täglichen Ziele ein)

	<p>Beginn der Projektzeit. Branch anlegen. Workflow-Tests Erstes Brainstorming und Spiel spielen Zieldefinitionen</p>
	<p>Fertigstellung der Planung bis hierhin  Nachmittag: Weiteres Brainstorming alleine, Klassendiagramme, etc. Nachholung des Storyboards</p>
	<p>Montag: Zusammenschmeißen aller Planungen und Cherry-Picking Erstes Anlegen besprochener Strukturen, Klassen, Methoden... --&gt; Struktureller Grundaufbau Aufgaben aufstellen und Verteilen (Trello) (Beginn Implementierung, wenn noch Zeit ist)</p>
	<p>Konkrete Implementierungen Aufteilung weiterer Arbeiten für den Nachmittag Unabhängige (asynchrone) Bearbeitung (Beispiel, jeder zwei Richtungsmethoden)</p>
	<p>Vormittag: Zusammenführung / Testen des Codes Besprechen der Ergebnisse des vorherigen Nachmittags  Dokumentation und Präsentation (am liebsten auch schon vormittags)</p>
	<p>Vormittag: Puffer</p>

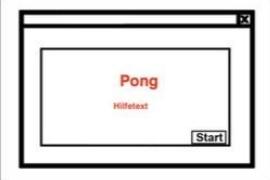
Präsentation &  
Abgabe

## Gruppen-Projekt: Spielimplementierung

### Beispiel für ein Storyboard für das Spiel „Pong“

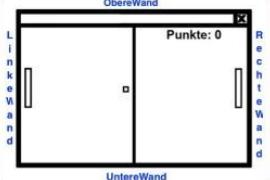
# Storyboard

**Start**



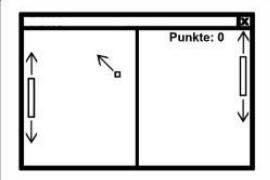
Das Spiel startet mit einem Titel und einem Hilfetext. Es gibt einen Startbutton bzw. eine Möglichkeit Pong zu starten.

**Das Spielfeld**



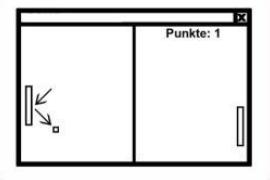
Es wird durch Wände definiert und hat eine Mittellinie. Auf dem Spielfeld befindet sich ein rechteckiger Ball und zwei Schläger. In der oberen rechten Ecke steht der Punktestand.

**Bewegung**



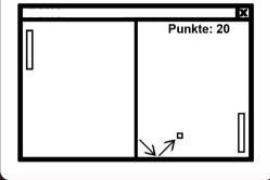
Der Ball beginnt in der Nähe der Mittellinie sich diagonal zu bewegen. Der rechte Schläger wird vom Computer gesteuert. Den linken kann die Spielerin mit Hilfe der Maus hoch und runter bewegen.

**Treffer Spielerschläger**



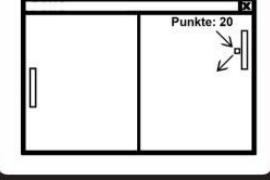
Trifft der Ball auf den Spielerenschläger erhöht sich der Punktestand um 1. Dabei ist der Einfallswinkel gleich dem Ausfallwinkel.

**Treffer Wand**



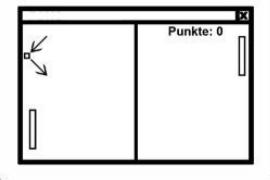
Trifft der Ball eine Seitenwand, prallt dieser wieder von der Wand am.

**Treffer Computerschläger**



Der Computerschläger trifft den Ball immer. Er kann nicht verlieren. Die Punkte erhöhen sich nicht.

**Treffer Spielerwand**



Trifft der Ball die Wand der Spielerin, reduziert sich der Punktestand auf 0. Das Spiel geht weiter.

Beim Storyboard geht es eine erste Visualisierung die die wichtigsten Funktionen einmal bildlich darstellt. Bei der sog. 'crazy eight technique' nimmt man dabei in der Regel die 8 wichtigsten Funktionen. Um jedem Bild nur einer Funktion zuzuordnen zu können, musste ich ein weiteres 9. Bild hinzufügen. Das liegt noch im Rahmen der Technik.