



### Aufgabe 3 Computerspiel 'Shut the Box'

Das Computerprogramm 'Shut the Box' soll für zwei Spieler entworfen werden. Mit diesem Programm sollen Kinder das Addieren von ganzen Zahlen spielerisch erlernen. Das Bild 1 zeigt die Spieloberfläche mit den Spieldaten ❶, den Klappen ❷, den Tasten 'würfeln' und 'Spielerwechsel' ❸ sowie der Würfelanzeige ❹. Die Klappen sind von 1 bis 9 durchnummeriert. Jeder Klappe ist der Wert zugeordnet, der auf ihr angezeigt ist. Im Bild unten hat ein Spieler bereits die Klappe 8 geschlossen, somit wird auf dieser nur ein 'X' angezeigt.

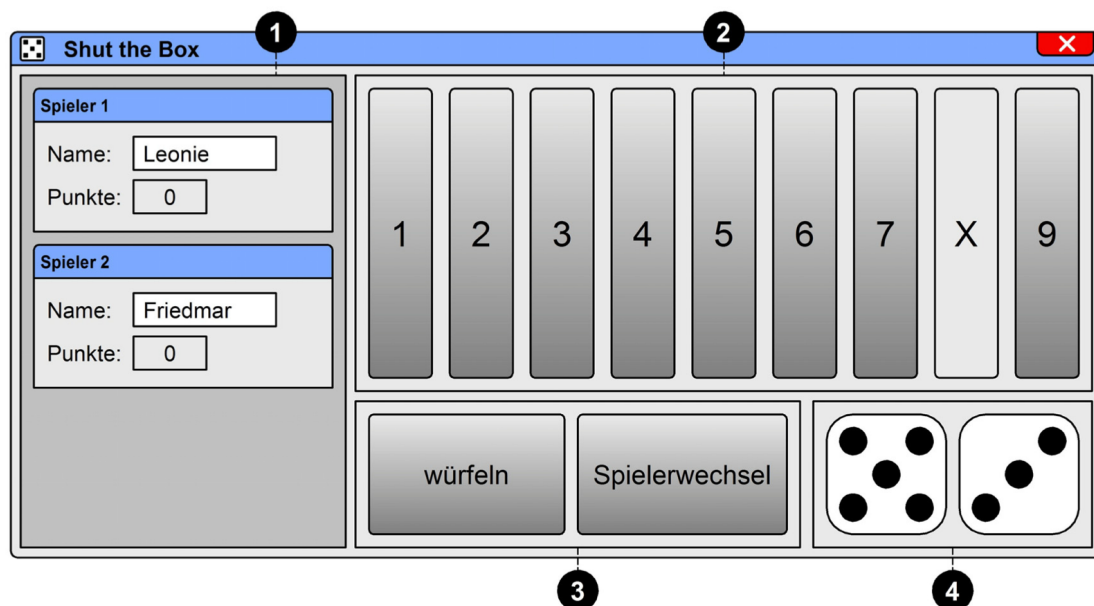


Bild 1

**Spielziel:** Ziel des Spiels ist es, alle Klappen zu schließen.

**Spielstart:** Alle Klappen von 1 bis 9 sind geöffnet, die Werte (1 - 9) aller Klappen werden angezeigt. Die Punktezähler der zwei Spieler sind auf Null gestellt.

**Spielverlauf:** Der Startspieler klickt auf die Taste 'würfeln' und generiert damit zwei Zufallszahlen, die mit der Würfelanzeige dargestellt werden. Die Würfeltaste wird deaktiviert. Nun muss der Spieler eine oder mehrere Klappen anklicken. Dabei muss die Summe der auf den Klappen angegebenen Werte mit der Augensumme der Würfel übereinstimmen. Die Klappen werden nach dem Anklicken geschlossen und sind danach nicht mehr anklickbar. Der Spieler entscheidet, auf welche Klappen er die Augensumme der Würfel aufteilt. **Beispiele:**

Augenzahlen	mögliche Klappen, die geschlossen werden können
1 + 1	2
1 + 2	1 + 2 oder 3
2 + 2	1 + 3 oder 4
2 + 3	1 + 4 oder 2 + 3 oder 5
3 + 3	1 + 5 oder 2 + 4 oder 1 + 2 + 3 oder 6

Hat der Spieler durch das Schließen der Klappen die passende Augensumme erreicht, wird die Würfeltaste wieder aktiviert und er kann erneut würfeln. Die bereits geschlossenen Klappen bleiben geschlossen. Ist es dem Spieler nicht möglich, die Augensumme der Würfel auf die noch offenen Klappen aufzuteilen oder sind alle Klappen geschlossen, so ist der Zug beendet. Er kann nur noch auf die Taste 'Spielerwechsel' klicken. Die Summe der Werte auf den noch offenen Klappen wird dem Spieler als Minuspunkte angerechnet und auf der Punkteanzeige dargestellt. Nun ist der nächste Spieler am Zug, alle Zahlenklappen werden wieder geöffnet. Kann auch dieser Spieler die Augensumme der Würfel nicht mehr auf die noch offenen Klappen aufteilen oder hat er alle Klappen geschlossen, so ist das Spiel beendet. Der Spieler mit den wenigsten Minuspunkten hat gewonnen.

**Beispiel:** Spieler 1 würfelt die Augensumme 9, siehe Bild 2. Es sind nur noch die Klappen 1, 3, 4 und 7 offen. Mit diesen Klappen kann die angezeigte Augensumme nicht summiert werden. Spieler 1 klickt die Taste 'Spielerwechsel', die Werte der noch offenen Klappen ( $1 + 3 + 4 + 7 = 15$ ) werden anschließend dem Spieler 1 als Minuspunkte angerechnet. Danach werden für Spieler 2 alle Klappen wieder geöffnet und er kann würfeln.

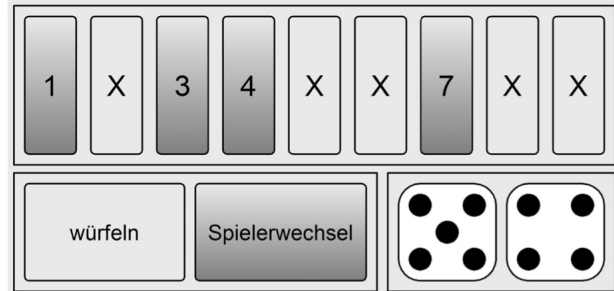


Bild 2

### 3.1 Analyse des Klassendiagramms

- 3.1.1 Der Name der Spieler darf während eines Spiels nicht mehr geändert werden. Erläutern Sie anhand der Klasse *Spieler* (Arbeitsblatt 1, Seite 5), wie diese Vorgabe umgesetzt wird. 2
- 3.1.2 Erläutern Sie den Begriff Datenkapselung anhand der Klasse *Spieler*. 1

### 3.2 Klassendiagramm ergänzen

Das Sequenzdiagramm im Bild 3 wurde bereits für die folgenden Szenarien entwickelt:

- Spieler 1 klickt auf die Taste 'würfeln' und würfelt die Zahlen 3 und 5.
  - Danach klickt Spieler 1 auf die offene Klappe mit der Zahl 8.
- Mögliche Nebenszenarien sind nicht eingezeichnet.

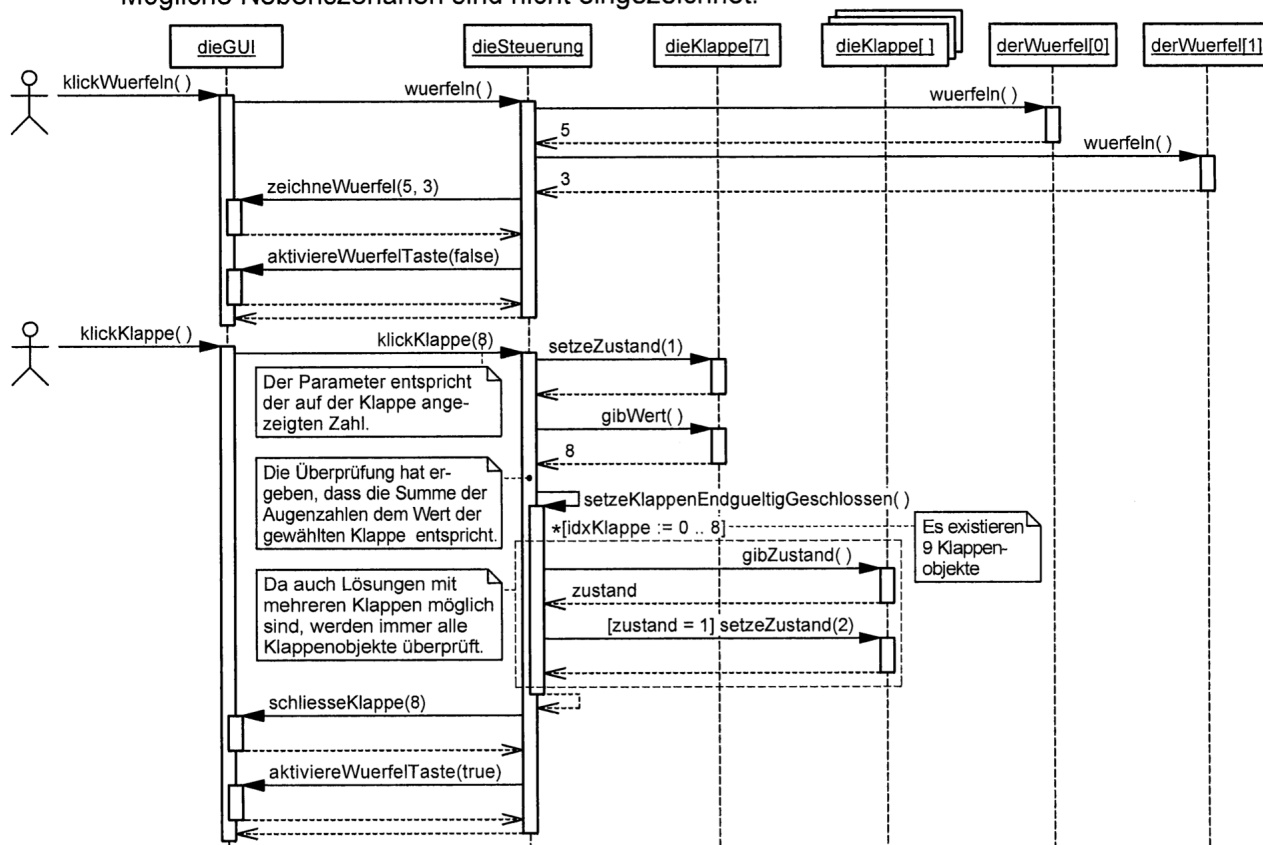
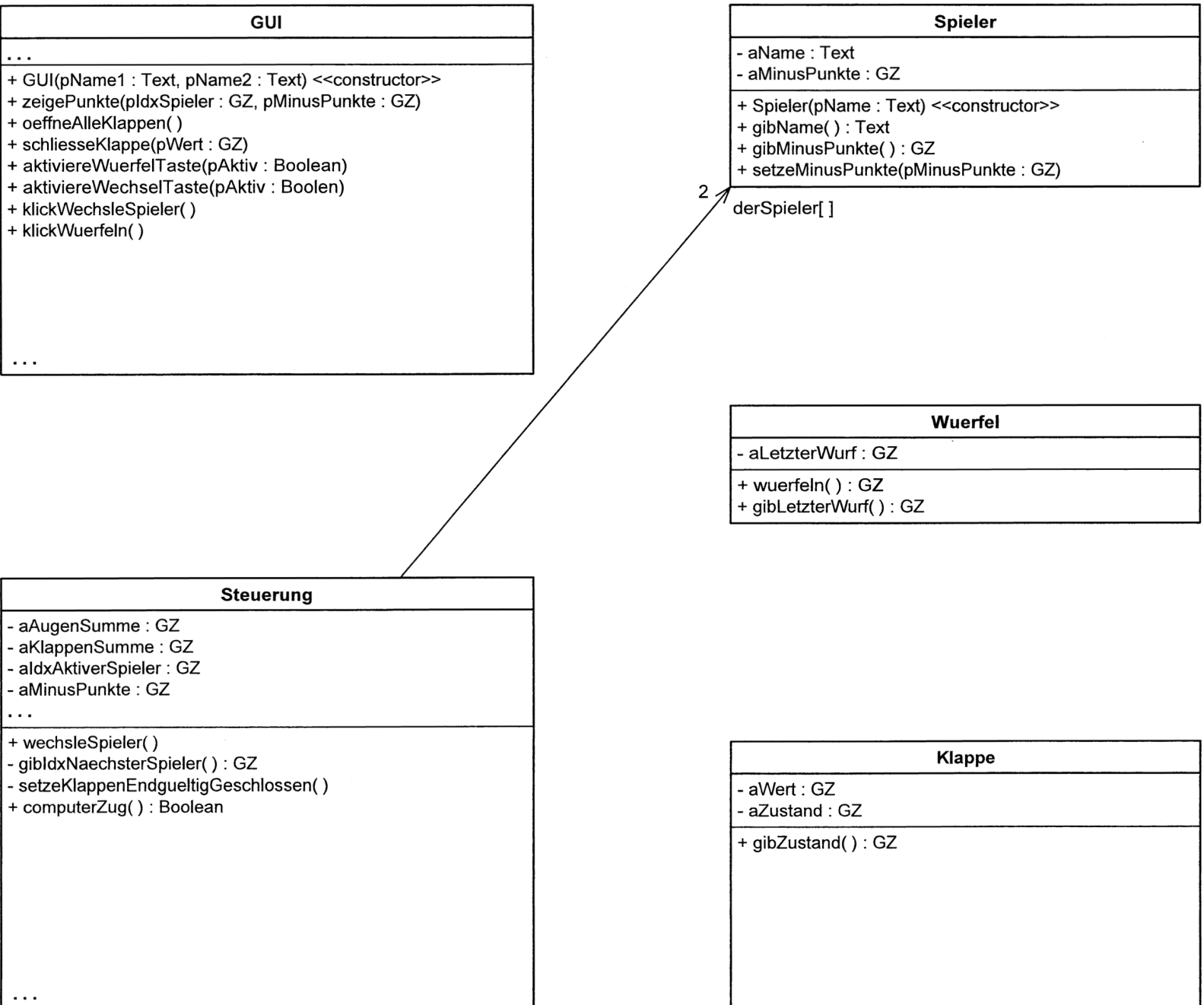


Bild 3

**Zum Verständnis:** Klappenobjekte besitzen drei Zustände. Offene Klappen haben den Zustandswert 0. Wurde eine Klappe geschlossen und der Spielzug ist noch nicht vorbei ( $\sum \text{Klappenwerte} < \sum \text{Augenzahlen}$ ), erhält diese den Zustandswert 1. Damit wird sichergestellt, dass keine falschen Werte in die Punkteberechnung einfließen. Eine endgültig geschlossene Klappe erhält den Zustandswert 2.



- |              |  | <b>Punkte</b> |
|--------------|--|---------------|
| <b>3.2.1</b> | Ein Objekt der Klasse <i>Steuerung</i> erhält bei seiner Erzeugung eine Referenz auf die GUI sowie die beiden Spielernamen. Die Objekte der Klasse <i>Klappe</i> erhalten bei ihrer Erzeugung ihren Wert (1..9). Ergänzen Sie auf dem Arbeitsblatt 1 die Konstruktoren der Klassen <i>Steuerung</i> und <i>Klappe</i> mit vollständiger Signatur und Sichtbarkeit.   | 2             |
| <b>3.2.2</b> | Überprüfen Sie, ob alle Botschaften des Sequenzdiagramms (Bild 3) im Klassendiagramm auf dem Arbeitsblatt 1 angegeben sind. Ergänzen Sie auf dem Arbeitsblatt die im Klassendiagramm fehlenden Operationen. Dabei ist die vollständige Signatur, der Rückgabotyp und die Sichtbarkeitskennzeichnung anzugeben. Die Sichtbarkeit öffentlich ist nur dann zu vergeben, wenn das für die Kommunikation zwischen den Objekten notwendig ist. | 3             |
| <b>3.2.3</b> | Ergänzen Sie auf dem Arbeitsblatt 1 die Assoziationen, die aus dem Sequenzdiagramm im Bild 3 ersichtlich sind. Die Assoziationen sind mit Navigationsrichtungen, Rollennamen und Kardinalitäten anzugeben. Begründen Sie Ihre Wahl der Richtung und Kardinalität der Assoziation.  | 3             |





### 3.4 Sequenzdiagramm aus PSEUDOCODE ableiten

5

Ergänzen Sie das Sequenzdiagramm auf nachfolgendem Arbeitsblatt für das Szenario „*Spieler 1 kann keine weiteren Klappen mehr schließen und klickt auf die Taste 'Spielerwechsel'*“. Für den aktiven Spieler 1 gilt: *dieSteuerung.aldxAktiverSpieler = 0*. *Spieler 1* hat 21, *Spieler 2* hat 25 (Minus-) Punkte. Für *Spieler 1* erhöhen sich für dieses Szenario die Punkte auf 36. Leiten Sie das zu erstellende Sequenzdiagramm aus dem Pseudocode der Operation *Steuerung.wechsleSpieler()* ab. Tragen Sie, soweit möglich, für die Parameter und Rückgabewerte der Botschaften im Sequenzdiagramm konkrete Werte ein.

OPERATION wechsleSpieler()

FUER idxKlappe <- 0 BIS 8 SCHRITT 1

WENN dieKlappe[idxKlappe].gibZustand() != 2

    aMinusPunkte <- aMinusPunkte  
        + dieKlappe[idxKlappe].gibWert()  
    dieKlappe[idxKlappe].setzeZustand(0)

ENDE WENN

ENDE FUER

aMinusPunkte <- aMinusPunkte  
    + derSpieler[aIdxAktiverSpieler].gibMinusPunkte()

derSpieler[aIdxAktiverSpieler].setzeMinusPunkte(aMinusPunkte)

dieGUI.zeigePunkte(aIdxAktiverSpieler, aMinusPunkte)

aIdxAktiverSpieler <- gibIdxNaechsterSpieler  
aMinusPunkte <- derSpieler[aIdxAktiverSpieler].gibMinusPunkte()

dieGUI.oeffneAlleKlappen()  
dieGUI.aktiviereWechselTaste(false)  
dieGUI.aktiviereWuerfelTaste(true)

