



# Projektdokumentation Modul 326

Autor\*in: [SteinhardtI@bzz.ch](mailto:SteinhardtI@bzz.ch)  
[Schaerern@bzz.ch](mailto:Schaerern@bzz.ch)

Klasse: IA20b

## Inhaltsverzeichnis

Projektbeschreibung .....	2
Anwendungsentwurf .....	2
Testfälle .....	3
Lösungsbeschreibung .....	4
Deklaration Eigenleistung .....	4
Anhang .....	4

## Abbildungsverzeichnis

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

## Tabellenverzeichnis

Tabelle 1: <b>Muster Anwendungsfallspezifikation</b> .....	3
--	---

## Änderungsverzeichnis

**Beschreiben Sie hier jeweils, was Sie wann geändert haben – insbesondere, wenn Sie bereits abgeschlossene Kapitel überarbeiten.**

DATUM	ÄNDERUNG
-------	----------

--	--



## Projektbeschreibung

Es handelt sich um eine Applikation, die das allbekannte Spiel Memory abbildet. Das Spiel wird in Java und für die Grafische Oberfläche mit swing umgesetzt. Es wird mit 20 verdeckten Karten gespielt und ein zufällig gewählter Spieler, macht den ersten Zug. Das Spiel kann entweder alleine gegen eine KI oder mit mehreren Spielern gespielt werden. Es wird entschieden ob auf der Konsole oder dem GUI gespielt wird, und am Ende werden die Punkte gezählt und der Gewinner gewählt. Weitere Verbesserungen könnten zb. eine KI mit verschiedenen Schwierigkeitsstufen etc. sein

## Vision und Ziele

*Unsere Ziele sind die Muss Aufgaben alle fertig zu stellen bis zum Ende des Moduls, dass die Komplettierten Aufgaben gut funktionieren und keine Fehler aufweisen. Wenn möglich und genügend Zeit vorhanden ist, wollen wir uns auch noch den Kann Aufgaben widmen.*

Muss	Kann
1 Menschlicher Spieler KI die Random aufdeckt Vorhandenes Grid 20 vorhandene Karten Auswahl zwischen Console und GUI Endet, wenn alle Karten aufgedeckt sind die Karten müssen klickbar sein. (click method) die Karten müssen drehbar sein. (Flip Method) die Karten müssen verglichen werden. (compare Method) Finaler Punktstand ersichtlich + Gewinner	Punkte werden gezeigt, während des Spiels. Neustart Knopf. Highscore = (meiste Paare) Auswahl, ob solo oder nicht. Verschiedene Schwierigkeitsgrade Spezialkarten

## Anwendungsentwurf

*Erstellen Sie in den nachfolgenden Unterkapiteln die notwendigen Artefakte (z.B. Textbeschreibungen von Anforderungen, UML-Diagramme, Mockups der grafischen Oberfläche), die Sie zur Erstellung der Anwendung benötigen. Darstellungen, welche grösser sind als der zur Verfügung stehende Platz, müssen als Vorschaubild dargestellt, im Anhang hinzugefügt und nachvollziehbar referenziert werden (bspw. «siehe Anhang 1 – Klassendiagramm Anwendung (Bild1.jpg)»).*

### Anwendungsfalldiagramm(e)

*Erstellen und beschreiben Sie hier ein Anwendungsfalldiagramm mit mindestens einem Anwendungsfall und mindestens einem Akteur.*

## Anwendungsfallspezifikation(en)

Erstellen Sie hier für jeden Anwendungsfall eine Anwendungsfallspezifikation.

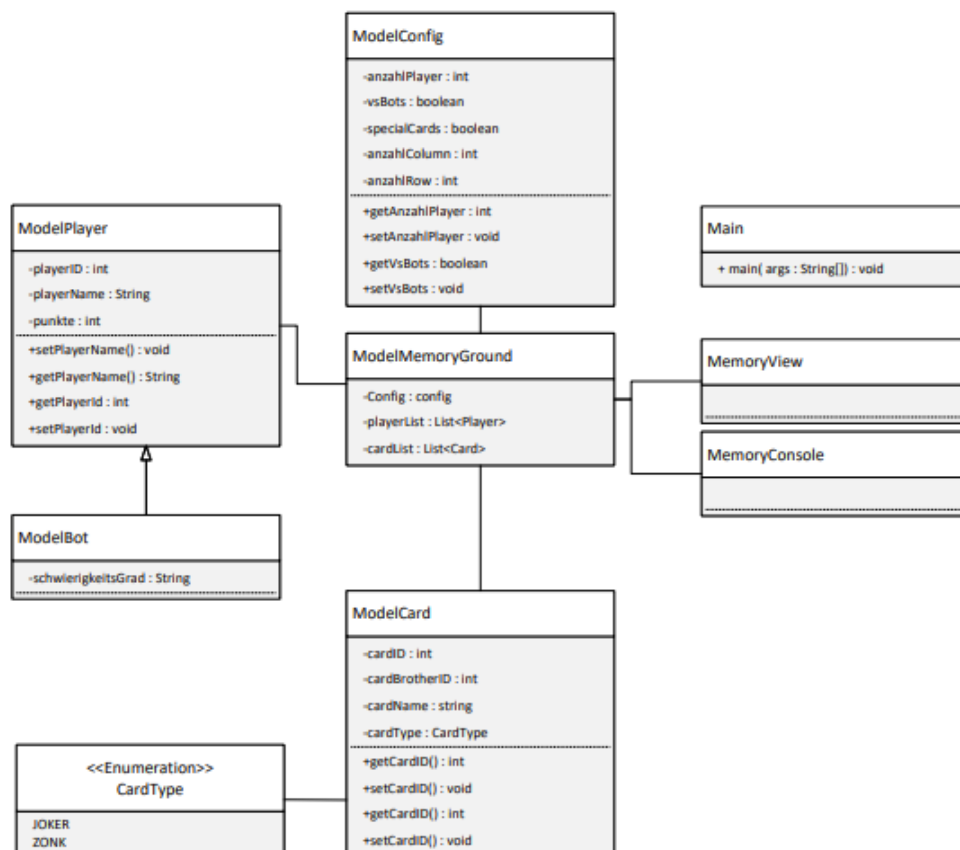
Tabelle 1: Muster Anwendungsfallspezifikation

ID	
Name	
Akteur(e)	
Auslöser	
Ablauf	1. 2.
Ergebnis	
Sonder- / Ausnahmefälle	

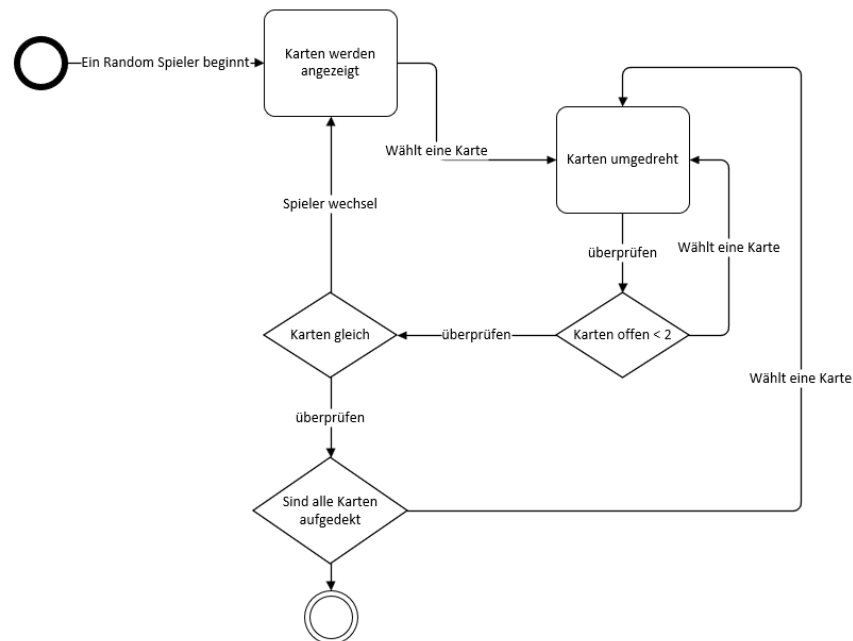
## Aktivitätsdiagramm(e)

Erstellen Sie hier mindestens ein Aktivitätsdiagramm für den gesamten Spielablauf inklusive der einzelnen Züge/Aktionen und beschreiben Sie diese nachvollziehbar.

## Klassendiagramm(e)



## Zustandsdiagramm(e)



## Mockup(s)

Erstellen Sie Mockups (Entwürfe der grafischen Oberfläche) zu Ihrem Spiel und beschreiben Sie diese nachvollziehbar.

## Testfälle

Beschreiben Sie in den Unterkapiteln «JUnit-Testfälle» sowie «Manuelle Testfälle» die Testfälle, welche Sie verwenden, um die fehlerfreie Arbeitsweise Ihrer Anwendung nachzuweisen.

Wichtige Elemente sind:

- Eindeutige Testfallnummer und Kurzbezeichnung
- Beschreibung des Testfalls (Klare Unterscheidung, ob allgemein, Konsolen- oder GUI-Version)
- Testdaten mit erwartetem Testresultat



*Am Ende müssen Sie die Durchführung der Testfälle angemessen und nachvollziehbar protokollieren. Die Protokolle der Testfälle sind dem Anhang hinzufügen.*

### **JUnit-Testfälle**

*Beschreiben Sie hier JUnit-Testfälle (mind. 10) gemäss obigem Schema, die Sie umsetzen werden.*

### **Manuelle Testfälle**

*Beschreiben Sie hier manuelle Testfälle (mind. 5) gemäss obigem Schema, die nicht automatisiert werden können und die Sie manuell ausführen werden.*

### **Lösungsbeschreibung**

*Beschreiben Sie hier, wie Ihre Lösung aussieht und was sie kann respektive was noch nicht umgesetzt wurde (z.B. wichtige Hinweise zum Programmcode). Bewerten Sie dazu die von Ihnen gesetzten Ziele und kommentieren Sie den Unterschied zwischen Ihrem IST-Ergebnis und dem ursprünglich definierten SOLL-Zustand. Ziehen Sie zum Schluss ein Fazit zu diesem Projekt, insbesondere was für Sie die wichtigste(n) Erkenntnis(se) ist/sind.*

### **Deklaration Eigenleistung**

Mit der Abgabe dieses Dokuments bestätige und versichere ich, dass ich die vorliegende Projektarbeit, das bedeutet sowohl die Dokumentation inklusive Anhänge als auch das Programm (Quellcode in Java), selbstständig, das heisst ohne fremde Hilfe, und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe.

### **Anhang**

*Hinweis: Anhänge müssen mindestens in einem Format vorhanden sein, welches ohne Zusatzprogramme dargestellt werden kann. Zu den akzeptierten Formaten gehören Bilddateien (JPEG, PNG, GIF), gängige Microsoft Office Dateien, PDF-Dokumente und Java-Dateien.*

*Folgende Anhänge werden erwartet:*

- Testprotokolle
- Source Code Programm
- Optional: Weitere Dateien