

## 1 Teilnehmer/innen des Teams:

Klasse: BI	Team: Kevin
---------------	----------------

## 2 Anforderungsdefinition (Meilenstein A)

„Balloon Pop“	
<b>Auftrag:</b> (Allgemeine Beschreibung)	<b>Szenario:</b> <ul style="list-style-type: none"><li>Es soll mit einem Vogel steigende Ballons kaputt gemacht werden, wenn nicht „Game Over“</li></ul>
<b>MUSS Kriterien:</b> (Konkrete Features, die umzusetzen sind)	<b>Folgende Features sollen implementiert werden (Funktionalität):</b> <ul style="list-style-type: none"><li>Vogel ist steuerbar von links nach rechts und umgekehrt</li><li>Ballons fliegen hoch</li><li>Vogel platzt Ballons</li><li>„Game Over“ wenn einer vorbei flieht</li></ul>

<b>KANN</b> <b>Kriterien:</b> (Konkrete Features, die optional sind)	<b>Folgende Features können zusätzlich implementiert werden: (Kreativität)</b> <ul style="list-style-type: none"> <li>• Zufällige Platzierung / schneller werden der Ballons</li> <li>• anzeige der geplatzten Ballons</li> </ul>
--	---

## 2.1 Planung LB2

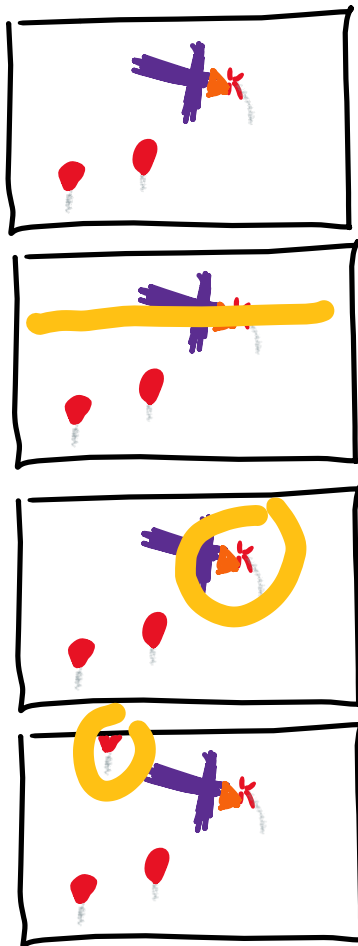
<i>MS</i>	<i>Tätigkeit / Abgabe</i>	<i>Soll-Datum</i>	<i>Ist-Datum</i>
A	<b>Projektstart</b> <ul style="list-style-type: none"> <li>➤ Team Bildung</li> <li>➤ <b>Wahl / Ausarbeitung der Anforderungsdefinition</b></li> </ul> Abnahme Anforderungsdefinition durch Lehrperson		
B	<b>Teamaufgabe 1:</b> <ul style="list-style-type: none"> <li>➤ <b>Abgabe: Lösungsdesign</b> (Analyse, Design: Funktionsmodell, UseCase, GUI, Storyboard)</li> </ul>		
B2	<b>Teamaufgabe 2:</b> <ul style="list-style-type: none"> <li>➤ Abgabe: Testvorschrift und Testfälle</li> </ul>		
C	<b>Einzelaufgabe 3:</b> <ul style="list-style-type: none"> <li>➤ Abgabe Szenario (.zip) mit Inline-Dokumentation, Systemdokumentation (UML Klassen-, Sequenzdiagramm)</li> <li>➤ <b>Fachgespräch Projektanbahnung</b></li> </ul>		
C2	<b>Einzelaufgabe 4:</b> <ul style="list-style-type: none"> <li>➤ Abgabe: Ausgefüllter Systemtest</li> </ul>		

### 3 Lösungsdesign (Meilenstein B: Teamaufgabe 1)

Anhand der Analyse wurde folgendes Lösungsdesign entworfen:

#### 3.1 Funktionsmodell

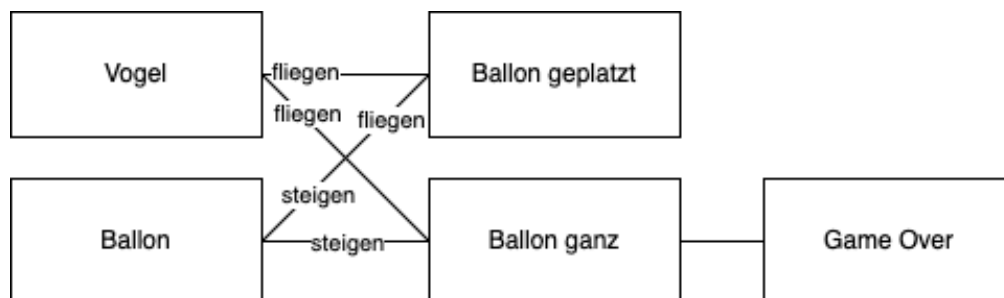
Im Folgenden sind die erwarteten Eingaben und Ausgaben beschrieben / dargestellt:



← →  
bewegen  
zerstören  
„Game Over“

#### 3.2 Anwendungsfälle (UseCases)

Folgende Anwendungsfälle sind hier detailliert dokumentiert:



### 3.3 Ablauf

Aus Benutzersicht ist folgender Ablauf des Programms zu erwarten:

Spiel starten	Spiel startet und beginnt
Pfeiltasten drücken	Vogel bewegt sich nach links oder rechts (am Rand geht es nicht weiter und stoppt)
Vogel berührt Ballon	Ballon platzt
Ballon gelangt oben an den Rand	Spiel ist vorbei und wird neu gestartet

## 4 Testvorschrift (LB2 Meilenstein B2: Teamaufgabe 2)

Testbeschreibung und vorbereitetes Testprotokoll siehe Dokument  
[M226B\\_LB2\\_Testvorschrift\\_MS-B2.docx](#)

## 5 Systemdokumentation (Meilenstein C: individuelle Aufgabe 3)

Das erstellte Java-Projekt (Greenfoot-Szenario) ist hier detailliert abgelegt:

[M226B\\_Aufgabe\\_3\\_Szenario\\_IhrName.zip](#)

### 5.1 Statisches Design: Klassendiagramm

Folgend die statische Struktur des Szenarios

...

(UML Klassendiagramm mit Assoziationen und Kardinalitäten)

### 5.2 Umfang / Abgrenzung / Änderungen gegenüber Design

Aufgrund unten beschriebener Umstände sind Anpassungen des ursprünglichen Lösungsdesigns gemacht worden:

...

(Umstände / Anpassungen / Veränderungen)

### 5.3 Funktionalität der Implementation.

Zusätzlich zu der Inline-Dokumentation sind hier folgende Funktionen detailliert beschrieben:

...

(Ausführliche Beschreibung der internen Funktionen  
oder Verweis zum Inline-Kommentar mit JavaDoc! (`/** @param @return **/`))

### 5.4 Dynamische Struktur: Sequenzdiagramm

Ein zentraler Ablauf eines UseCases ist im Folgenden dargestellt:

...

(Darstellung eines zentralen Ablaufs mittels Sequenzdiagramm)

**Trace: ...**

...

## 6 Bedienungsanleitung (**Meilenstein C: individuelle Aufgabe 3**)

...

## 7 Testprotokoll (**LB2 Meilenstein C2: individuelle Aufgabe 4**)

Ausgefülltes Testprotokoll siehe Dokument  
[M226B\\_LB2\\_Testvorschrift\\_MS-C2\\_Name.docx](#)