



SEP Informatik – Wintersemester 2016/2017

Pinball: Simulation eines Flipperautomaten Lastenheft

Matthias Dangl, Armin Größlinger, Janet Siegmund

1 Eine kurze Bemerkung vorab

Dies ist **Euer** Praktikum. Dieses Dokument ist kein Katalog von Aufgaben, der Punkt für Punkt abgearbeitet werden muss, um das SEP zu bestehen, sondern lediglich eine Reihe von Hinweisen, was wir erwarten. Es ist aus diesem Grund knapp gehalten. Wie **Euer** Programm letztendlich aussehen soll, müsst **Ihr** selbst entscheiden.

2 Motivation

Ein Pinnball-Spiel vereint mehrere für Informatiker interessante Herausforderungen: Zum Einen muss das Model-View-Controller-Prinzip umgesetzt werden, um die eigentliche Logik (Physik und Spielregeln) von der Oberfläche zu entkoppeln. Dies ermöglicht enorme Flexibilität und Erweiterbarkeit. Eine zweite Herausforderung stellen die kontinuierlichen physikalischen Berechnungen und die parallel dazu laufend aktualisierte GUI mit den damit verbundenen Multi-Threading-Anforderungen dar. Daher ermög-

licht die Erstellung des Spiels das Kennenlernen von unterschiedlichen Technologien (Multi-Threading, Kommunikation, GUI Programmierung, Design Patterns).

3 Aufgabenstellung

Ziel des SEP im Wintersemester 2016 ist es, eine Simulation eines Flipperautomaten (auch Pinball-Spiel genannt) zu entwickeln. Mehrere Spieler sollen gemeinsam (abwechselnd) an einem PC Pinball spielen können. Das Programm soll dabei die physikalischen Gegebenheiten annähernd genau simulieren, die Pinball-Regeln auswerten und für jeden Spieler die Punktezahl übernehmen.

Das Programm soll das aktuelle Spiel grafisch anschaulich darstellen. Dabei müssen das Spielfeld inklusive etwaige Spielelemente, Hindernisse und Besonderheiten, die Kugel sowie gegebenenfalls Zusatzinformationen, z.B. der Punktestand, dargestellt werden. Außerdem muss der Spieler den Flipper bedienen können.

Nutzer sollen die Möglichkeit haben, in einem grafischen Editor selbst neue Spielfeld-Anordnungen anzulegen und zu speichern und existierende Anordnungen zu laden und zu verändern.

Die Anwendung muss mit Java 7 oder Java 8 und Swing entwickelt werden. Es ist besonders darauf zu achten, das Model-View-Controller-Pattern konsequent umzusetzen. Ebenfalls sollen andere wichtige Design Patterns, z.B. das Observer Pattern, bei der Implementierung berücksichtigt werden.

4 Produkteinsatz

Zielgruppe der Anwendung sind Personen, die gemeinsam spielen möchten. Auch der Einsatz bei Turnieren im Freundeskreis oder z.B. im Rahmen von Veranstaltungen von Hochschulgruppen soll möglich sein.

5 Produktfunktionen

5.1 Minimale Leistungsmerkmale

5.1.1 Spiel

- Modell:
 - Verwaltung des aktuellen Spielzustands
 - Auswertung der physikalischen Gesetze während der Bewegung der Kugel in einer zweidimensionalen Ebene, in der die Schwerkraft nach unten hin wirkt
 - Unterstützte Spielelemente: Schlagtürme (Bumper), Steinschleudern (Slingshots), Zielscheiben, Rampen, runde und eckige Hindernisse, Löcher (Holes) & Kicker

- Durch Flächen mit besonderen Beschleunigungseffekten sollen Rampen trotz des zweidimensionalen Berechnungsmodells simuliert werden können.
- Regelauswertung mit Punktezählung, Zählung verbleibender Kugeln, etc.
- Trennung der Regelauswertung von der Physik
- Fünf mitgelieferte Spielflächen, die sich an den üblichen Flipperautomaten orientieren und insgesamt alle Spielelemente abdecken
- Laden von abgespeicherten Spielfeldern (Format freigestellt, Empfehlung: serialisierte Objekte)
- GUI:
 - Darstellungsfenster, das mindestens bei der Auflösung von 1024x768 Bildpunkten benutzt werden kann
 - Möglichkeit, Spiel mit zu wählender Anzahl an Spielern (min. 8) zu starten
 - Darstellung des laufenden Spiels (Spielfeld, Kugel, Flipper, Punktestand, evtl. Zusatzinformationen)
 - Eingabemöglichkeiten für erlaubte Aktionen während des Spiels (Flippersteuerung, ggf. Stoßen)
 - Die GUI muss den Benutzer sinnvoll unterstützen und benutzerfreundliche Eingabeelemente anbieten.
 - Flüssige Darstellung
 - Unabhängigkeit vom Regelwerk und der Physik

5.1.2 Spielfeld-Editor

- Anlegen, laden und speichern von Spielfeldern, die im Spiel verwendet werden können
- Unterstützung aller Spielelemente, die auch das Spiel unterstützt
- Benutzerfreundliche Bedienung (mit Drag-and-Drop von Spielelementen)

5.2 Optionale Leistungsmerkmale

Folgende Ideen können als Anstoß für eigene Erweiterungen des Programms dienen:

- Weitere Spielelemente (z.B. Elektromagnete, Tore, etc.)
- Missionen (wie z.B. in *Full Tilt! Pinball*¹)
- Möglichkeit, Kugeln („Leben“) dazuzuverdienen
- High-Scores für Spieler über mehrere Spiele hinweg

¹https://en.wikipedia.org/wiki/Full_Tilt!_Pinball

- Statistiken zu Spielen
- Mehrsprachenfähigkeit
- KIs

6 Organisatorisches

Für jede Phase des Praktikums muss ein Phasendokument abgegeben werden. Dieses Dokument ist Grundlage für das Kolloquium am Ende jeder Phase, in dem der Phasenverantwortliche die Ergebnisse der Phase vorträgt. Die genauen Termine entnehmt bitte der Homepage².

7 Bewertung

Die Benotung des SEP richtet sich nach folgenden Kriterien:

- Qualität der abgegebenen Dokumente
- Qualität der Kolloquien und der Individualleistung in wöchentlichen Meetings
- Qualität der Abschlusspräsentation
- Qualität des Source-Codes
- Erfüllung der minimalen Leistungsmerkmale (s.o.)
- Sinnvolle Erweiterungen über diese Merkmale hinaus
- Robustheit des erstellten Programms

Diese Liste hat keine Reihenfolge, die einer Gewichtung entspricht. Es gibt sicherlich weitere Punkte, die als selbstverständlich gelten und sich bei Nichterfüllen negativ auswirken (z.B. sollte die Bedienung möglichst einfach von der Hand gehen). Umgekehrt wirkt sich eine besonders gute Erfüllung einer Anforderung natürlich positiv aus.

Viel Erfolg und vor allem viel Spaß!

²<http://www.sosy-lab.org/Teaching/2016-WS-SEP/>