**Software-Entwurf**

**Prüfungsstudienarbeit zur Vorlesung „Java Programmierung“**Prüfer: Prof. Dr. Andreas Berl

**Teilnehmer 1: Michael Mican, 00692390, michael.mican@stud.th-deg.de, AI**

Name, Matrikelnummer, Email, Studiengang

**Teilnehmer 2: Philipp Muhr, 00692390, philipp.muhr@stud.th-deg.de, AI**

Name, Matrikelnummer, Email, Studiengang

**Semester:** SS 2019

**Datum der Abgabe:**

# ASCIIPinball

**Inhalt:**

* Ein **Mock-Up** mit ausführlicher Erklärung aller Features.
* Ein **grobes Klassendiagramm** mit ausführlicher Erklärung.
* Weitere geplante Ansätze.

**Umfang: ca. 2 Seiten**

1. **Mock-Up**

Ball

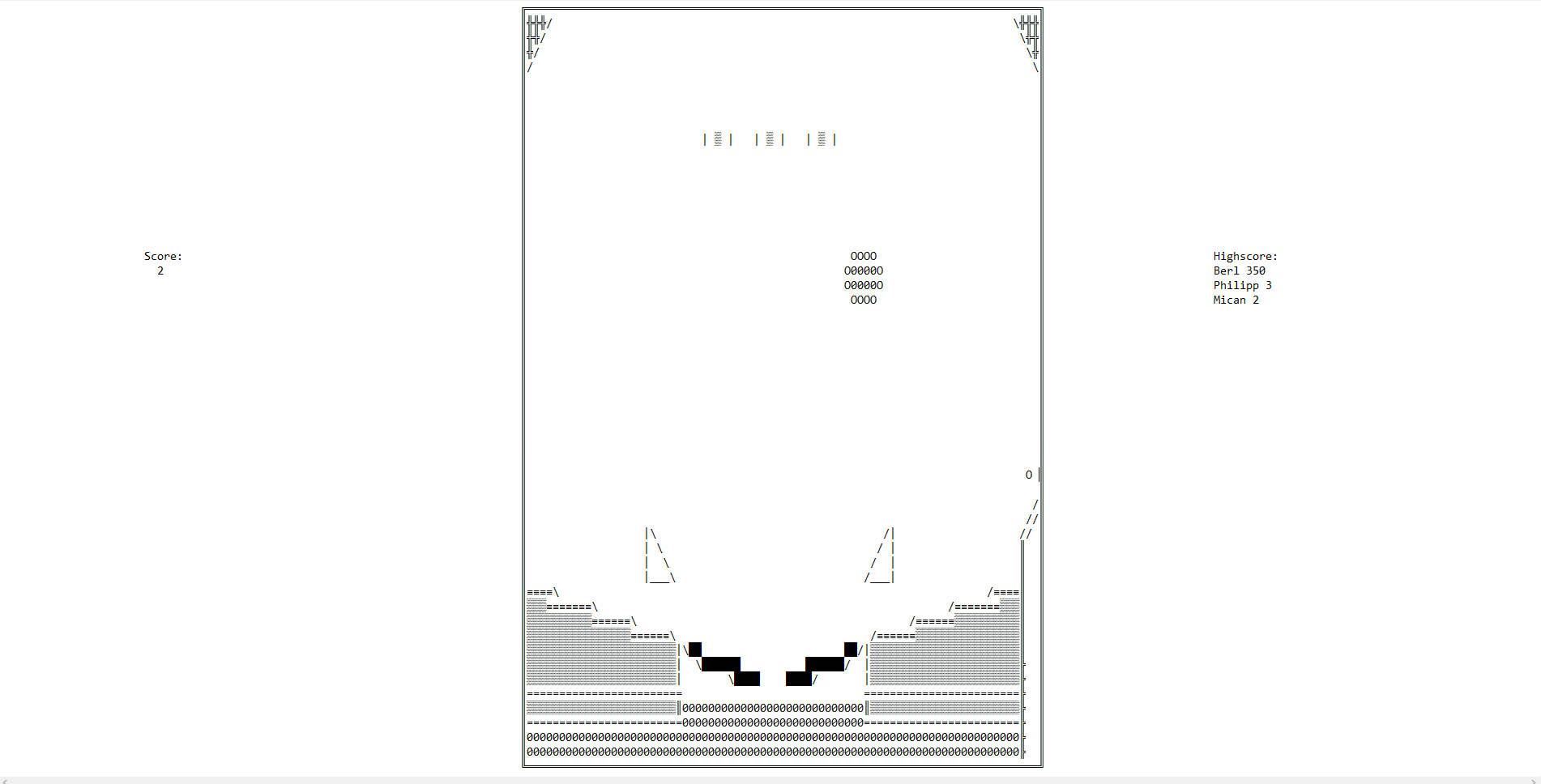
Highscoreliste

Bumper

Point-Door

Aktueller Score des aktuellen Spielers

Klassische Wände



Das Spiel versucht einen einfachen Flippertisch nachzubilden.

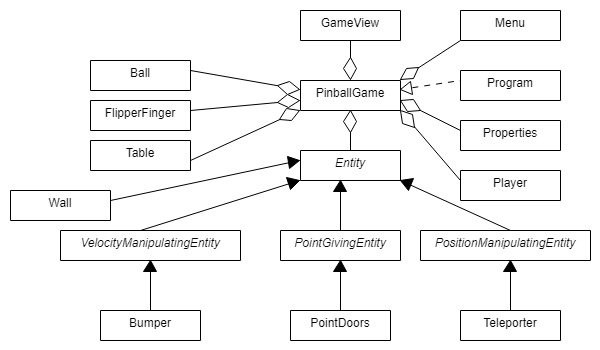
Am Anfang des Spiels hat der Nutzter die möglichkeit in einem Menü zwischen 1 – 4 Spielern auszuwählen und diesen einen Namen zu geben. Diese wechseln sich Reih um ab, sobald der vorherige Spieler eine seiner drei Runden verloren hat.

Der Ball verfügt natürlich über eine Physik und prallt von Objekten ab. Außerdem wirkt die Gravitationskraft auf Ihn. Es gibt verschiedene Hindernisse, manche versetzen den Ball (z.B. Teleporter flächen), andere geben dem Spieler bei „überrollen“ Punkte (z.B. Point-Doors) und wiederum andere schubsen den Ball aktiv Weg (z.B. Bumper).

Der Spieler versucht nun den Ball möglichst Punkte bringend über das Spielbrett zu schießen. Hierfür hat er zwei Flipperfinger welche er separat, links und rechts, ansteuern kann.

Im Spiel enthalten ist außerdem ein Scoreing system, das den Spieler für das Treffen verschiedener Objekte (Entities) mit Punkten entlohnt. Diese Punkte entscheiden bei Mehrspielerspielen am ende des Spiels über den Sieg. Außerdem werden die Besten Scores in einer Highscore-Top-10-Liste mit Namen des Spielers verewigt.

1. **Klassendiagramm**



Herzstück des Programs ist die Klasse „PinballGame“ hier werden alle „einzelteile“ des Spieltisches zusammengeführt. Außerdem sind hier alle aufzurufenden Methoden, welche von der Klasse Program nurnoch in einer while(true) schleife in der richtigen reihenfolge zum richtigen Zeitpunkt aufgerufen werden. Das PinballGame ebthällt mindestens einen Ball, welcher seine bewegung und die wirkende Gravitationskraft verwaltet (nicht die Physikalische kollision! Diese wird von den Entities verwaltet). Die Klasse Table baut die Grundlegende Box bzw. das Grundlegende Design des Tisches auf und verwaltet die Physik dieser Äußeren Wände. Außerdem besitzt das Spiel mindestens 2 Flipperfinger. In dieser Klasse werden die Interaktionen des Spielers mit den Flipperfingern, sowie die Physikalische Interaktion zwischen Flipperfinger und Ball gesteuert.

Außerdem besteht das Spiel noch aus verschiedenen Entities. Dies ist eine Abstracte klasse und gibt Ihren unterklassen bereits Variablen (wie xpos und ypos), sowie einige Methoden (wie interactWithBall()) für die unterklassen zu implementierung vor. Diese vorgaben werden von den 3 weiteren Abstrakten Unterklassen (VelocityManipulatingEntity, PointGivingEntity und PositionManipulatingEntity), je nach Typ der Entity erweitert. Von den Abstrakten Klassen erben nun die Klassen, welche die tatsächlichen hindernisse Representieren. Aktuell sind erstmal Wall (Begrenzende Wände ohne besondere Eigenschaft), Bumper (Hinderniss, dass den Ball beschleunigt, wenn er von Ihm abprallt), PointDoors (Scorepunkt gebende flächen ohne Kollision) und Teleporter (Ball versezende Objekte) geplant. Dieser Aufbau wurde gewählt um eine möglichst leichte erweiterung der vorhandenen Entities zu ermöglichen.

Die Klasse Properties ist eine Klasse mit statischen Variablen. Diese wird von der Menu Klasse bei erstmaligen Spielstart gesetzt und dient als Einstellungsdatei (z.B. Einstellung wieviele Spieler spielen gerade am Tisch, oder wie hoch ist die gravitation).

Wie bereits erwähnt wird die Menu Klasse bei start des Spieles verwendet. Hier kann der Spieler auswählen mit wievielen Leuten er spielen möchte.

Die Klasse Player verwaltet die Scores der Spieler. Am Ende des Spiels werden die Instanzen dieser Klasse an die Klasse HighScore gegeben, welche dann das Leaderboard managed. Außerdem wird die Klasse die Top 10 auf der Festplatte speichern bzw. bei Spielstart laden.

1. **Ausblick / In Planung**

Zu zwecken des einfachen Level Designs ist der Plan eine LoadLevel Methode in PinballGame zu implementieren, welche am Anfang des Spiels aus einer auf der Festplatte gespeicherten lvl.txt datei einen Spieltisch zu Generieren. Somit wird es dem nutzer möglich sein einen eigenen Spieltisch (in einem Texteditior seines vertrauens) zu entwerfen und diesen Anschließen auch zu Spielen.