

Technisches Dokument: Arcade Table

von Nikolai Stecker, Sabrina Thomsen und Florian Völkers

im Rahmen der Fächer Mobile Systeme / IT-Systeme

der Studiengänge Media Systems und Medientechnik

Konzept

Da wir während der Umsetzung unseres geplanten Produkts realisiert haben, dass dieses mit unseren Mitteln nicht umsetzbar ist und wir daher ein neues Produktkonzept entwickelt haben, würden wir dieses zunächst etwas erklären und anschließend auf die technischen Details eingehen.

Beim Arcade Table handelt es sich um einen digitalen Spieletisch, auf dem man klassische Arcade-Spiele, wie Pong, Air-Hockey oder Tron, im direkten Duell gegeneinander spielen kann. Die Spiele werden dabei durch verschiedene Smartphone-Inputs gesteuert. Pong ist der Sportart Tennis nachempfunden und Ziel ist es den Ball am Schläger des Gegners vorbei zu schlagen. Air-Hockey ist der Sportart Ice-Hockey nachempfunden und das Ziel ist es den Puck in das Tor des Gegners zu schlagen. Bei Tron fährt man mit zwei Motorrädern auf einem Spielfeld und hinterlässt eine Spur. Das Ziel ist es nicht in die eigene oder die Spur des Gegners zu kommen.



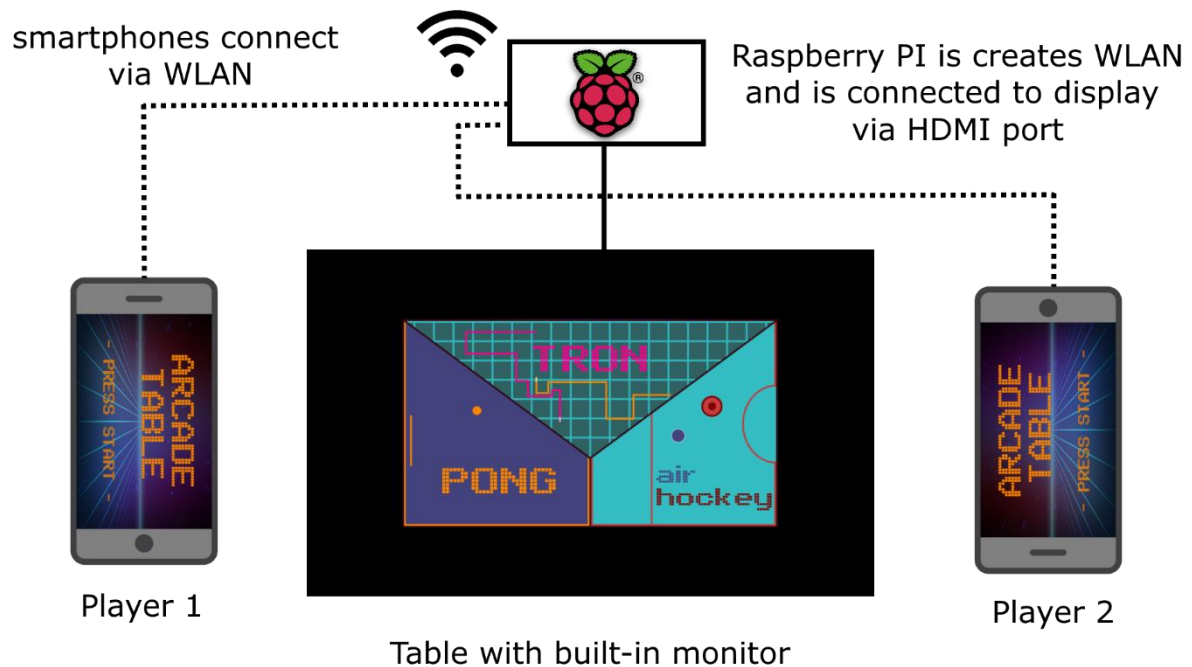
Abbildung 1: Abschlusspräsentation und Ausstellung der Games Master Hamburg 2017. © 2017 HAW / Games-Master / Fotografin: Bente Stachowske



Technischer Aufbau

Das Herzstück des Arcade Table ist der Raspberry PI, der alle Komponenten miteinander verbindet. Der Raspberry PI erstellt ein eigenes WLAN, dient also als Server, und verarbeitet die Spiellogik. Über einen HDMI Eingang wird der Raspberry PI mit dem im Tisch verbauten Monitor verbunden. Dort werden die Spiele dargestellt. Beide Spieler verwenden als Controller ein Android-

Smartphone, das über WLAN mit dem Raspberry PI verbunden ist. Auf dem Smartphone werden die verschiedenen Input-Daten gesammelt und an den Raspberry PI gesendet. Dort werden diese verarbeitet und die Spiellogik wird umgesetzt.



Technische Umsetzung

Die Entwicklung des Arcade Table haben wir in drei Aufgabenbereiche aufgeteilt. Das Erstellen der App, das Erstellen der Spiele und des Servers auf dem Raspberry PI und das Erstellen des Tisches selbst. So konnten wir jeder zeitgleich an verschiedenen Aufgaben arbeiten.

Der Tisch

Für den Tisch diente ein günstiger Couchtisch von Ikea (90 x 55 cm) als Grundlage. In diesen wurde eine passende Einlassung für einen 27 Zoll Monitor hineingesägt. Damit der Monitor im Tisch hält, wurden 3 Holzlatten zurecht gesägt und von unten an den Tisch geschraubt. Nun konnte der Monitor eingelassen und die Monitorkabel durch die Lücken der Holzlatten geführt werden. Zum Schutz des Monitors wurde auf den Tisch eine Plexiglasplatte (90 x 55 cm) geklebt. Außerdem wurde diese so mit einer schwarzen, glänzenden Folie beklebt, dass nur der Display zu sehen bleibt.



Abbildung 2: Der Arcade Table ohne eingebauten Monitor und Plexiglasplatte.

Die App

Für die Entwicklung der App entschieden wir uns zur Verwendung von Android Studio. Zwar konnte so nur für Android Smartphones entwickelt werden, jedoch war sowieso von Anfang an geplant die Smartphones für die Besucher zu stellen.

Für die Kommunikation mit dem Server, dem Raspberry PI, wurden Sockets verwendet. Die Verbindung wird direkt in der MainActivity hergestellt, wenn der Spieler „Start“ drückt. Anschließend muss der Spieler auswählen, ob er Spieler 1 (links vom Tisch) oder Spieler 2 ist (rechts vom Tisch). Dies ist wichtig, da sich aufgrund dessen die Verwertung der Inputs unterscheiden. Zum Beispiel bedeutet im Spiel Pong ein Kippen des Smartphones vom User nach rechts, dass Spieler 1 seinen Schläger nach unten bewegt, jedoch Spieler 2 seinen Schläger nach oben bewegt.

Zum Senden der Input-Daten wird ein Thread verwendet, der entweder auf Befehl oder alle 50ms ein Datenpaket an den Server sendet. Um zu erkennen, wann das Spiel beendet ist, wird ein weiterer Thread verwendet, der darauf wartet, dass der Server das Endsignal sendet, ein String mit dem Inhalt „end“. Sobald dies geschieht, wird die jeweilige GameActivity beendet und man kehrt zurück ins Spielmenü.

Das Spiel

Das Menü und die Spiele wurden mit Zuhilfenahme der Bibliothek pygame entwickelt. Pygame vereinfacht die Darstellung und Bewegung von Bildern und Formen mit Python. Außerdem ermöglicht es die Abfrage von Kollisionen von gezeichneten Objekten.

Im Menü wird nach dem Starten ein Server-Thread ausgeführt, der auf die Verbindung von zwei Smartphones über Sockets wartet. Sobald dies geschehen ist, wird ein Player-Thread gestartet, der die Daten der einzelnen Spieler empfängt und verarbeitet. Die Auswahl des Spiels geschieht durch das Senden des Spielnamens von Spieler 1. Im Spiel selbst werden dann zunächst die Verbindungen konfiguriert und anschließend geht das Spiel in eine Schleife bis die Endbedingung (z.B. eine bestimmte Punktzahl) erreicht wird. In dieser Schleife werden die gesendeten Daten in Bewegungen umgesetzt, das grafische Interface der Spiele wird ständig geupdatet und eventuelle Kollisionen und Bedingung zum Spielende werden abgefragt. Sobald das Spielende erkannt wurde, wird das Signal „end“ an beide Smartphones verschickt, die Spielschleife wird beendet und man kehrt in das Hauptmenü zurück.

Um einen Überblick über die erstellte Software, die im Projekt verwendet wird, zu erhalten, haben wir uns entschieden die Kommunikation der verschiedenen Klassen und Methoden als Sequenzdiagramm darzustellen. Darin wird schematisch ein typischer Ablauf als Beispiel dargestellt. Dieser kann sich jedoch mehrmals wiederholen oder abgebrochen werden. Da das Sequenzdiagramm bei einer Darstellung aller Inhalte der Software viel zu umfangreich geworden wäre, werden nur die wichtigsten Abläufe dargestellt.

