

# **Dokumentation & Projekttagbuch**

Innovation Lab 2  
Jahr 2022

Projekt: **Raspberry PI Arcade Game**

Team: **Gruppe 26**

# 1. Allgemeine Informationen

**Projektname:** Raspberry PI Arcade Game

**Supervisor:** Markus Petz

Innovation Lab 1, Sommersemester 2022

**Projektteam:**

Nicholas Hoeller, [if19b043@technikum-wien.at](mailto:if19b043@technikum-wien.at)

David Xing, [if19b124@technikum-wien.at](mailto:if19b124@technikum-wien.at)

Marko Zepic, [if20b256@technikum-wien.at](mailto:if20b256@technikum-wien.at)

## Management-Summary des Projektes

Ziel des Projekts ist ein Spielautomat im Arcade-stil mit funktionstüchtigen Spielen. Hierzu bedarf es einer brauchbaren Programmierumgebung für Retrospiele (Libretro/Retroarch) bzw. Cores ("Plugins" für Libretro API). Im Anschluss soll die Software auf einem Raspberry PI in einem Arcade Automaten verbaut werden und Studierenden bereitgestellt werden. Des Weiteren soll dieses Projekt als Grundlage für weitere Semesterarbeiten dienen.

**Projektschwerpunkte:**

- geeignete Programmierumgebung ermitteln (richtet sich aktuell an Kenntnisstand der Gruppe!)
- simples Spiel als SW für die Umgebung implementieren (z.B. Pong)
- auf Raspberry PI portieren
- Hardware vom Automaten als Input für Raspberry PI implementieren
- Verbau des Raspberry PI in Automaten

## Rahmenbedingungen und Projektumfeld

Hinsichtlich der Rahmenbedingungen für das Projekt gibt es (noch) keine speziellen Angaben/Vorgaben. Es muss lediglich das Systemumfeld innerhalb der Gruppe ermittelt werden, mit dem es möglich ist, die gewünschten Ziele komfortabel und effizient umzusetzen.

## Semester-Roadmap

Aktuell ist geplant, dass Projekt in zwei Semester aufzuteilen. Für das vergangene Semester war die Implementierung eines Cores/Emulators für Libretro gedacht und erfüllt. Anschließend, wird im 4. Semester (INNO2) mithilfe dieses Cores ein Spiel entwickelt, welches dann auf einem Raspberry PI laufen soll und in einem Arcade Automaten verbaut wird.

**Grobe Roadmap für SS2022 (INNO2):**

- März/April: Ideen für ein Spiel sammeln und Plan dazu erstellen
- April/Mai: Grundgerüst & Basis-Funktionalitäten implementieren
- Mai/Juni: weitere Implementierungen; lauffähiges Spiel

Grobe Roadmap für WS2022 (INNO3):

- November: Informationen bzgl. Hardware sammeln & erste Implementierungsversuche
- Dezember: Planung und evt. schon Verbau der Hardware in einen Automaten
- Jänner: Wartung, Optimierung

## **Collaboration & Tooling**

Für das Projekt werden folgende Tools verwendet:

- GitHub: Repository für Version Control und Source Code Management (Link: <https://github.com/Miriaana/InnoLab-Raspberry-Pi-Arcade-Game>)
- evt. Trello: Aufteilung der Arbeitspakete (Kanban Board)

## **Anmerkungen**

Da wir zurzeit leider noch nicht wissen, wie viele Gruppenmitglieder jetzt mit Sicherheit noch am Projekt arbeiten, ist es durchaus möglich, dass es Abweichungen bei der obigen Roadmap gibt. Aktuell ist Marko Zepic (if20b256) das einzige Gruppenmitglied, das sicher dabei ist. David Xing (if19b124) und Nicholas Hoeller (if19b043) haben noch offene Nachprüfungen aus dem vorherigen Semester. Miriam Böhm (if20b066) hat sich letztes Semester geäußert, eigentlich noch gerne weiter an dem Projekt arbeiten zu wollen, jedoch kann sie ebenfalls noch nicht tatsächlich zur Gruppe dazugezählt werden.

## 2. Projekt-Kurzbeschreibung

Am Ende des Projekts soll ein Spiel vorhanden sein, welches sich im Core vom letzten Semester ausführen und spielen lässt. Dieses Gesamtpaket wird dann auch auf dem Raspberry PI laufen.

Die größten Herausforderungen werden höchstwahrscheinlich bei der eigentlichen Implementierung des Spiels sein, da wir auch hier wieder auf viel Recherche sowie konstantes Testen und Ausprobieren des Quellcodes angewiesen sind. Als User-Interface werden wir wieder RetroArch verwenden und die Implementierungen werden in Visual Studio stattfinden.

Zurzeit kann leider nicht viel zur Projektbeschreibung genannt werden, da uns noch der Wissensstand bezüglich den Projektzielen fehlt. Aus diesem Grund und auch weil sich unser Team doch ein wenig reduziert hat, haben wir vor, während des Projekts die Ziele und Anforderungen zunächst einmal simpel zu halten.

# 3. Spezifikation der Lösung

## TO BE EDITED IN THE FUTURE

*< Nach Abschluss der Auftragsklärung (Vorprojekt-Phase) starten Sie die Projektdurchführung. Erstellen Sie parallel zur Umsetzung Ihres Projekts über die Sprints hinweg eine Spezifikation Ihrer Lösung!*

*Vor jedem Sprint müssen mindestens jene Details spezifiziert sein, die Sie im kommenden Sprint umsetzen werden. Verwenden Sie hierzu Techniken wie zum Beispiel das Schreiben von Epics & User Stories und bauen Sie ein Product Backlog auf (wenden Sie die LV-Inhalte aus der LV Agiles Projektmanagement an).*

*Verwenden Sie zur Spezifikation generell Visualisierungstechniken, die zur Aufgabenstellung passen. Zum Beispiel können ergänzend zu den Mockups und User Stories auch Datenbankdiagramme, Klassendiagramme, oder Sequenzdiagramme (Darstellung zeitlicher Abläufe) sinnvoll sein.*

*Normalerweise geht man vom Groben zum Detail vor. Die Gliederung dieses Abschnitts kann wie folgt aussehen:*

- *Systemumfeld: Beschreiben Sie die Abgrenzung der umzusetzenden Lösung (Systemgrenzen)*
- *Features (Funktionale Anforderungen): Alle geforderten Lösungseigenschaften - bei Software i.d.R. die Features bzw. eine Beschreibung dieser als User-Stories o.ä.)*
  - o *Erstellen Sie Screen Mockups aller wesentlichen UI-Ansichten!*
- *Schnittstellen: Alle relevanten Schnittstellen Ihrer Lösung.*
- *Qualitätseigenschaften, technische Anforderungen (Nicht-Funktionale Anforderungen): Performance, Skalierbarkeit, Verfügbarkeit, Benutzbarkeit, Angaben zur Architektur und Erweiterbarkeit, etc.*
- *Sonstige „nicht auf den ersten Blick klare“ aber wesentliche Lösungsmerkmale!*

*Stimmen Sie mit Ihrer Betreuer\*in ab, wie die Spezifikation aufgebaut sein soll!*

*Fragen Sie nach, wann immer Sie das Gefühl haben, es könnte sich um ein Missverständnis oder unterschiedliche Erwartungen handeln oder wenn Sie eine Anforderung nicht ganz klar verstanden haben! >*

# 4. Aufwandschätzung

SIEHE EXCEL DATEI

*< Verwenden Sie die erklärte Delphi Methode, um den Aufwand für diese Semester zu schätzen, schreiben Sie hier die Ergebnisse erklärend rein und verweisen Sie auch auf das verwendete Excel Dokument. >*

# 5. Auslieferung

Die fertige Lösung des Projekts soll auf ein ausführbares Spiel sein, welches mithilfe von RetroArch in unseren Core hineingeladen werden kann. Für die Implementierung wird wieder die Header-Datei von libretro verwendet und bzgl. Ausführung wird das Spiel als eine DLL-Library kompiliert .Für die Handhabung auf einem Raspberry PI werden laut unserem aktuellen Wissensstand jedoch andere Dateiformate benutzt.

# 6. Unser Projekt-Tagebuch

## TO BE EDITED IN THE FUTURE

*< Dieser Abschnitt solle eine Art Tagebuch sein, wo Sie festhalten „was jeweils bei uns im Team passiert ist im Projekt“. Verwenden Sie Fotos aus Ihren Meetings, fotografieren sie etwaige Überlegungen von Whiteboards. Machen Sie Screenshots.*

*Beschreiben sie in kurzen Textabschnitten welche Probleme es gab, welche Herausforderungen gelöst wurden, was „cool“ war im Projekt etc.*

*ACHTUNG: Erstellen sie diesen Abschnitt laufend (!) parallel zum Projekt und nicht erst am Ende am letzten Abend vor der Projektabgabe! Dies ermöglicht Ihren Betreuer\*innen auch am Ende nachvollziehen zu können, warum etwas besonders gut oder eben nicht so gut funktioniert hat, warum es zu großen Fortschritten oder zu Verzögerungen kam, etc.*

*In der Praxis nutzt man ein solches Tagebuch als Grundlage für eine Projekt-Retrospektive und Team-Feedback-Runden.*

*Tipp: Treffen Sie einander am Ende des Semesters und lassen Sie ihr Projekt bei einem guten Projektabschluss-Essen „Revue passieren“: Dies ist eine gute Gelegenheit, um das Erlebte nochmals zu besprechen und für die Zukunft bzw. um das Gelernte in das nächste Semester und Innovation Lab mitzunehmen! >*