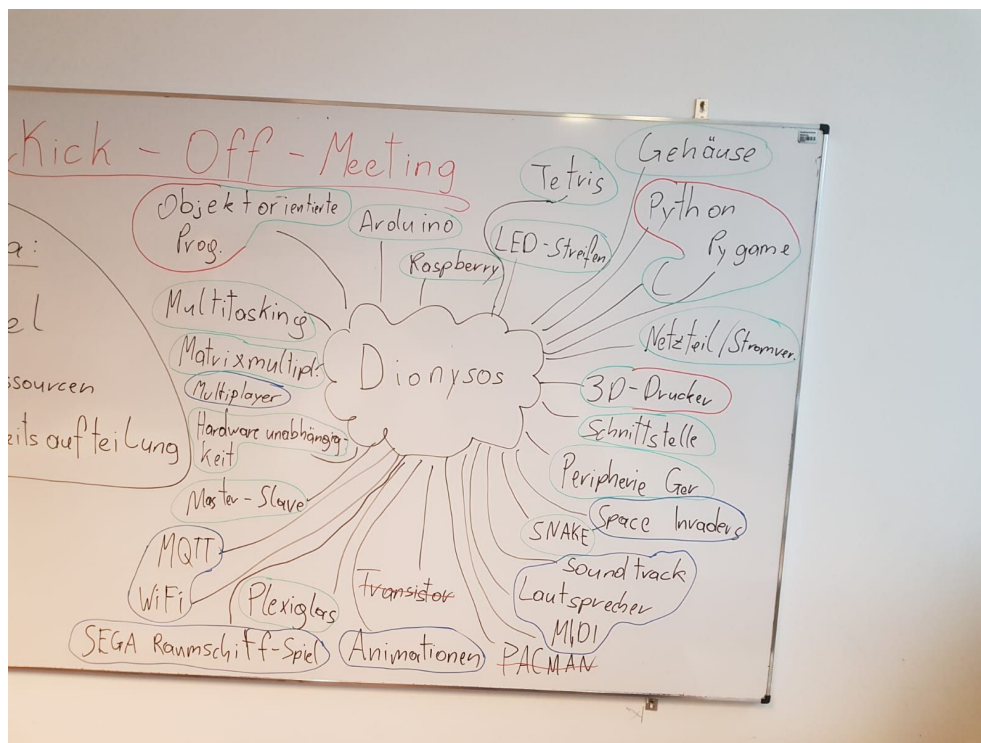


# Dionysos Projekt

## Ziel:

Entwicklung einer mikrocontroller basierten Matrix-Display Lösung mit Anbindung an einen Einplatinencomputer.

Das Display sollte eine Auflösung von 20 x 10 LEDs unterstützen und als Proof of Concept. würden wir das beliebte Retrospiel Tetris nutzen, welches durch einen Controller oder Tastatur gesteuert wird. Für ein ergonomisches Gehäuse wo alle Bauteile unterkommen, wird natürlich auch gesorgt. Zur Programmierung werden die Programmiersprachen Python und C eingesetzt (darunter auch einige Bibliotheken z.B. Adafruit Neopixel für die LED-Streifen) und Konzepte wie Objektorientierte Programmierung oder Funktionale Programmierung um den Code besser zu strukturieren und Lesbarkeit zu fördern.



## Konzepte:

- **Arduino:**  
Mikrocontroller für Steuerung des LED-Streifens. Umwandlung 2D Befehle in LED-Streifen Kommandos. Verbindungsaufbau zu Master.
- **Raspberry:**  
Backend Controller für Darstellungen (Audio),.... Peripherie Schnittstelle und Master des Arduino's.

**Projektteam:** Lohner Dominik (Gr. Bc), Hinterberger Florian (Gr. Bd)

- **LED-Streifen:**  
WS2812B (RGB) wird für die Darstellung genutzt.
- **Netzteil:**  
Stromversorgung der Elektronik
- **Peripherie Geräte:**  
Geräte die die Schnittstelle zum Menschen darstellen (Controller, Tastatur,...)
- **Gehäuse:**  
Zum Unterbringen des Großteils der Komponenten (praktisches Design). Plexiglas (milchig) für den Schutz der Augen und des LED Streifens.
- **Multitasking:**  
Parallelisieren von Aufgaben für den Raspberry/Arduino.
- **Master-Slave:**  
Raspberry gibt als Master Aufgaben an den Arduino Slave weiter.
- **Retrospiel(e):**  
Tetris, (Snake), (Sega Raumschiff), (Space Invaders)
- **Hardwareunabhängigkeit**  
→ Arduino schickt Bildschirmgröße an Master (Displayklasse)
- **Matrix Algebra:**  
→ für Darstellungen (Multiplikation/Bewegungsmatrizen)
- **3D-Druck:**  
für Gehäuseteile

Extras:

- **Animationen**
- **MQTT/WiFi** → **Schnittstelle Linux/Windows**  
→ **Multiplayer**
- **Sound, Lautsprecher, MIDI**

