



# Project Jump

Entwicklung eines Systems zur Sprungerkennung mittels ESP32-WROOM und ADXL345

# Inhalt

- Warum dieses Projekt?
- Systemarchitektur
- Komponentendiagramm
- Herausforderungen & Lösungen
- Fazit & Ausblick



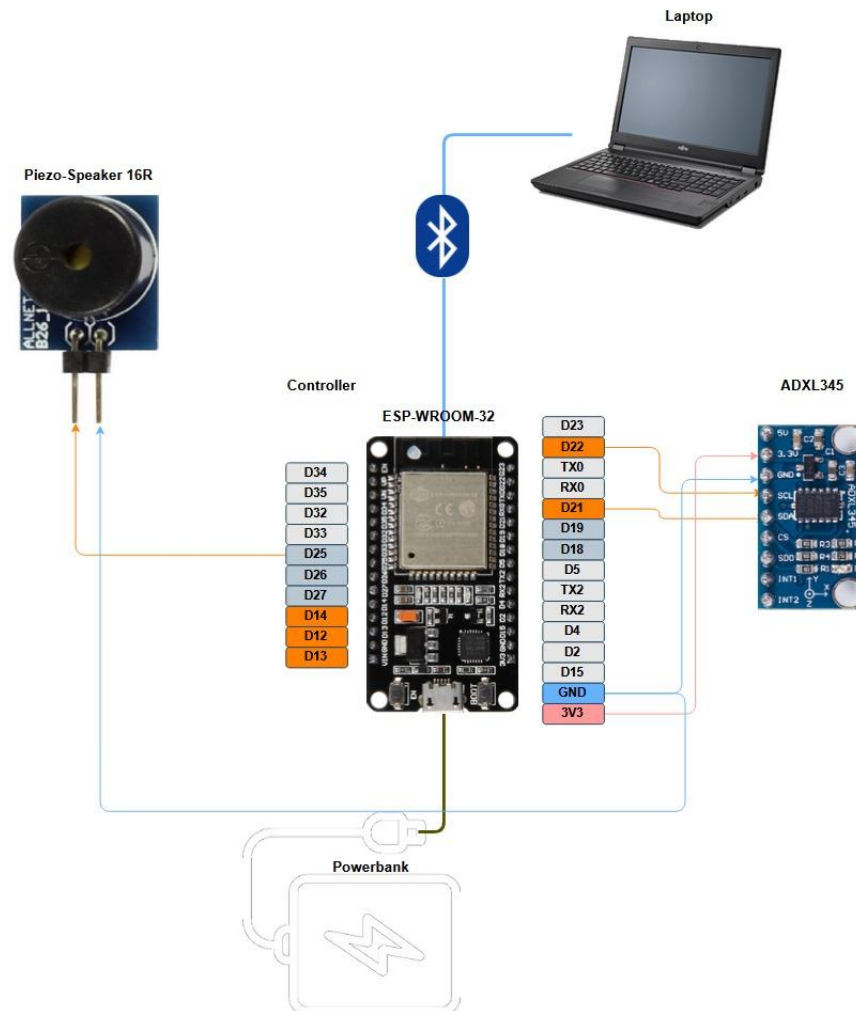
# Warum dieses Projekt?

- Herausforderung:
  - Automatische Sprungerkennung für interaktive Anwendungen
  - Realisierung einer einfachen Verbindung zwischen Hardware und Software
- Ziel:
  - Erkennung eines Sprungereignisses mittels ESP32-WROOM und ADXL345 und Simulation eines Tastendrucks der Leertaste zur Steuerung eines einfachen Spiels

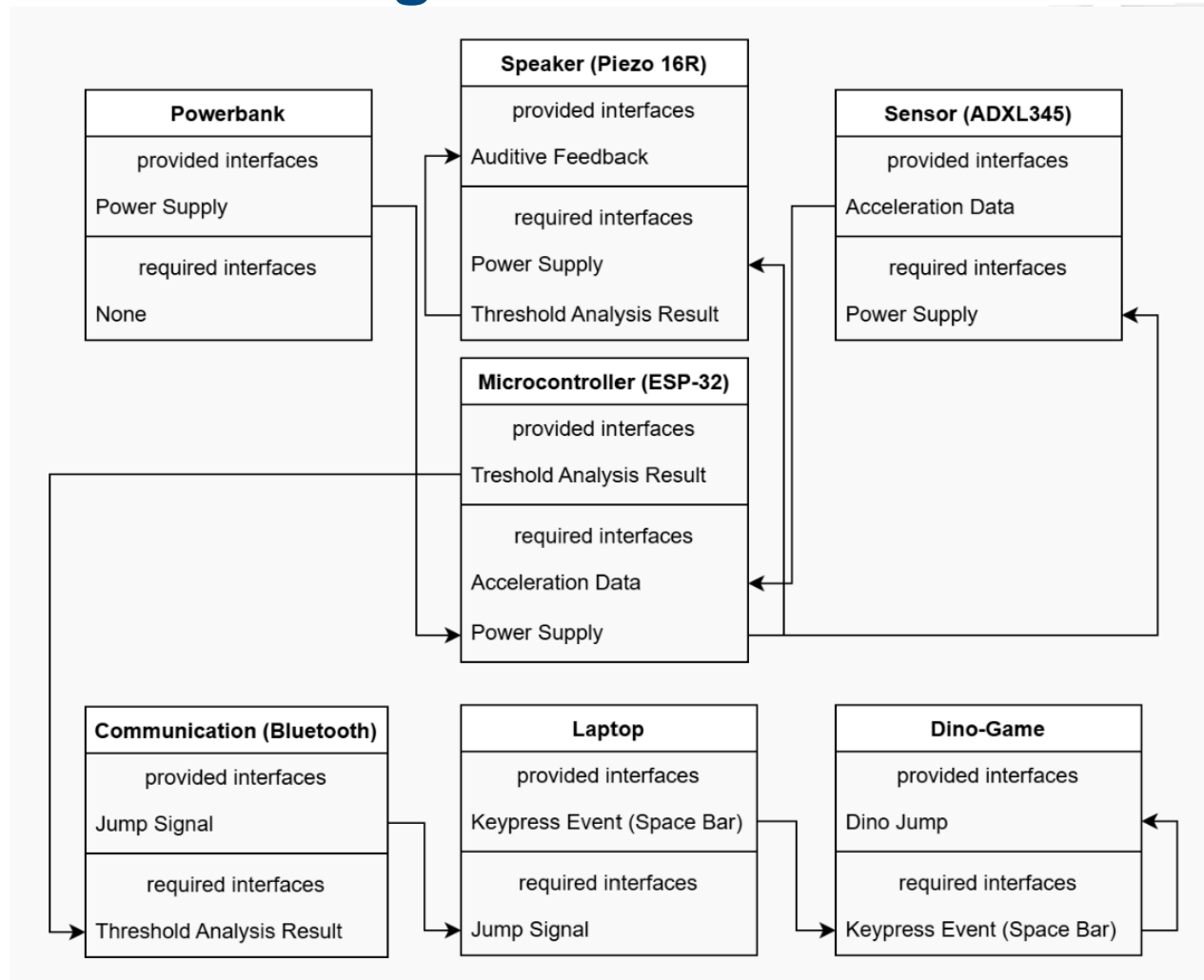
# Systemarchitektur

- Powerbank: Liefert Stromversorgung für ESP32-WROOM und ADXL345.
- Sensor (ADXL345): Misst die Beschleunigung.
- ESP32-WROOM: Analysiert die Daten und erkennt Sprünge mithilfe eines Schwellenwertes.
- Lautsprecher (Piezo-Speaker 16R): Gibt auditives Feedback bei Kalibrierung und Sprünge.
- Bluetooth-Kommunikation: Sendet „Sprung“-Signale an den Laptop.
- Laptop: Simuliert das Drücken der Leertaste mittels Python-Bibliothek.
- Spiel: Reagiert auf (simulierten) Tastendruck der Leertaste und lässt den Dino springen.

# Systemarchitektur



# Komponentendiagramm



# Herausforderungen & Lösungen

- Herausforderungen:
  - Zuverlässige Schwellenwertbestimmung
  - Verzögerung
- Lösungen:
  - Schwellenwertbestimmung mittels fünf Kalibriersprüngen
  - Testungen mit verschiedenen Softwarestrukturen

# Fazit & Ausblick

- Fazit:
  - Erfolgreiche Integration von Hardware und Software
  - Effiziente Sprungerkennung für interaktive Anwendungen
- Ausblick:
  - Verbesserung der Sensorgenauigkeit
  - Minimierung der Verzögerung
  - Integration von Doppelsprüngen („Leertaste gedrückt halten“)
  - Ducken (Vögel im Spiel)
  - Optimierung Gehäuse
  - Auf Anwendung des Steckbretts verzichten (Gehäuse verkleinern) & direkte Verbindungen löten





**VIELEN DANK FÜR IHRE AUFMERKSAMKEIT!**