

Hands On Session

AI-Spielplatz
6th of May 2025

Oliver Welz (ScaDS.AI)



What's it all about?

💡 Intro to Materials & Resources
(own) Setup and how to start



...no worries: it can be overwhelming first but we'll do this together and be sure you'll learn a lot



What we'll do in this 1st Hands On Session...

- Intro to general setup, resources, software
 - Login to PC, WiFi, Cloud-Folder
- Overview of software we'll use:
 - Msty (your connection to LLMs that are here to help)
 - Jupyter/Colab (place where you can program stuff)
 - GitHub Repository (here you'll find materials & resources)
- Overview over hardware we'll use:
 - What is a Raspberry Pi? How to start & connect via VNC...
 - How to apply a sensor?
 - Which Sensors do we have?
 - Audio
 - Camera



What we'll do in this 1st Hands On Session...

- Hands-On:
 - Let's build our first simple project together
 - Try or tutorials:
Beginners: [What is GitHub, Python?](#) [How to use Jupyter/Colab?](#) [What is an API?](#) Advanced [- Interact with OpenAIs API](#) [- Attach a Sensor to RPis GPIO](#)
- Dream Big:
 - First ideas of what you could do? Let's discuss!

Short Question

...how do you feel today [handsign:  |  | ],

...who tested ai and for what? 

Getting started...

Laptop Login: *open one of the Laptops and login please*
User: scadsai
Password: scadsai

WLAN: *check & change now and set to auto-connect*
SSID: ai-spielplatz
Password: ai-spielplatz

Cloud Link:
<https://cloud.scadsai.uni-leipzig.de/index.php/s/8TWEMX32KgfrrwF>



- Share Projects, Data, etc.
- Link expires end of May



Software we'll use:

What's prepared on the Laptops,
what could you install on your hardware...



Msty
Why Msty
Pro / Contra

 msty.app

What is Msty?

- Msty is a modern **AI-Chat-UI**, you can use to interact with diverse LLMs – locally as well as online/remote
 - we have a LLM-Engine installed locally that is called **ollama**
 - as LLM provider we use **OpenAI** (GPT-4o & 4o-mini are available)
- Simple usage of local and remote models
- **Offline-First, Online-Ready:** Msty is optimized for offline-usage
- **Great model compatibility:** compatible with models from HuggingFace, Ollama, Open Router, etc.
- **Data Privacy (offline mode!):** Personal information & chat data won't leave the device

Msty

Why Msty

Pro / Contra

[msty.app](#)

Pros

- ✓ **Flexible** – use local & remote AI-models
- ✓ **Data Privacy** (offline mode!) – Data won't leave the device
- ✓ **Features**, like **RAG** (Chat with Documents), **Websearch**, etc.

Cons

- ✗ **Freemium**: „always free“ Version for private usage and premium-subscriptions
- ✗ **Closed Source** – not an Open-Source-Project

<https://docs.msty.app>

Short Demo 
...repeating is learning 

Jupyter

Why Jupyter

Pro / Contra

 jupyter.org

What is Jupyter?

- Jupyter is a powerful **interactive coding environment**
- Works **locally on your machine** – perfect for offline experimentation
- Write and run **Python code in cells** with immediate output
- Great for **data analysis, machine learning, and AI prototyping**)

Pros

- ✓ **Full Control** – runs on your own system, no internet needed
- ✓ **Flexible** – integrate any local tools, models, or datasets
- ✓ **Beginner-Friendly** – clear cell-by-cell execution

Cons

- ✗ **Setup Required** – you need to install Python and Jupyter
- ✗ **Limited Collaboration** – not cloud-based unless manually configured

Google Colab

Why Colab

Pro / Contra

 colab.research.google.com

What is Google Colab?

- Colab is a **cloud-based Jupyter Notebook platform** by Google
- No installation needed – runs entirely in your **browser**
- Comes with **free GPU support** for AI/ML experiments
- Perfect for **sharing code, collaborating, and quick starts**

Pros

- ✓ **Instant Start** – works directly in the browser
- ✓ **Free Hardware Acceleration** – GPU/TPU for heavy AI models
- ✓ **Easy Collaboration** – share notebooks like Google Docs

Cons

- ✗ **Online Only** – requires internet access
- ✗ **Session Timeouts** – limited runtime for free version
- ✗ **Data Privacy** – data is stored on Google's servers

Short Demo 
...repeating is learning 

github.com

What is GitHub?

- GitHub is a **platform for sharing and managing code**
- Enables **version control** using Git – track changes, collaborate easily
- Ideal for **team projects, open-source sharing, and code backups**

Git-What?

Why GitHub

Pro / Contra

Pros

- ✓ **Collaboration-Friendly** – work together on shared code
- ✓ **Version History** – track every change over time
- ✓ **Public or Private** – control who sees your work
- ✓ **Integrated with Colab & Jupyter** – open notebooks directly

Cons

- ✗ **Learning Curve** – Git/GitHub can be confusing at first
- ✗ **Requires GitHub Account** – to contribute or manage projects

Our Repo: Ai Spielplatz

 github.com/IvaroEkel/AI-Spielplatz

Repository Structure [... important]

Warming-Up/

Introductory exercises to get started with AI concepts.

Tutorials/

Step-by-step guides for various (AI) projects.  most important to start

Projects/

Sample AI projects to explore and learn from.

Data/

Datasets used in tutorials and projects.

Other_Resources/

Additional materials and references. *Not relevant...*

Slides/

Presentation slides used during the workshop.  always check back here

website/

Files related to the workshop's informational website on GitHub. *Not relevant...*



Hardware we prepared:

What's **hardware** is prepared,
how to start where?





Hardware we prepared:

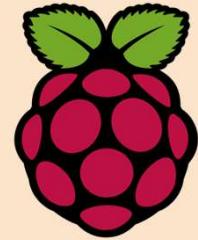
Disclaimer:

Don't worry: you can't break anything but if 💩 happens, it's no problem.

You are here to try & learn.



Also: we tried to prepare everything in detail but it happens that we missed something.
Feel free to inform us if anything isn't working.



Raspberry Pi?

What is it &
How to start...

🔗 [raspberrypi.com](https://www.raspberrypi.com)

🧠 What is Raspberry Pi?

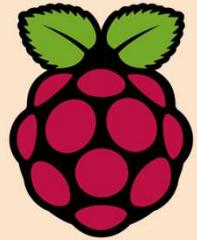
- Raspberry Pi is a **small, affordable single-board computer**
- Originally designed for **learning programming & electronics**
- Perfect for **AI experiments**, robotics, sensors, and creative projects
- Runs Linux (e.g. Raspberry Pi OS) – can be used like a tiny PC
- Can be used with Python, Jupyter, and local LLMs

✓ Pros

- ✓ **Low Cost** – powerful but super affordable
- ✓ **Beginner-Friendly** – huge community & tons of tutorials
- ✓ **Flexible Use** – AI, smart home, media server, etc.
- ✓ **Offline & Portable** – ideal for secure, mobile AI setups

✗ Cons

- ✗ **Basic Setup Needed** – install OS on SD card and connect peripherals
- ✗ **Limited Power** – not suitable for very large AI models or heavy compute tasks



Raspberry Pi?

What is it &
How to start...

🔗 raspberrypi.com

🧠 What is GitHub?

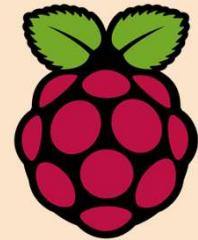
- GitHub is a **platform for sharing and managing code**
- Enables **version control** using Git – track changes, collaborate easily
- Ideal for **team projects, open-source sharing, and code backups**

✓ Pros

- ✓ **Collaboration-Friendly** – work together on shared code
- ✓ **Version History** – track every change over time
- ✓ **Public or Private** – control who sees your work
- ✓ **Integrated with Colab & Jupyter** – open notebooks directly

✗ Cons

- ✗ **Learning Curve** – Git/GitHub can be confusing at first
- ✗ **Requires GitHub Account** – to contribute or manage projects

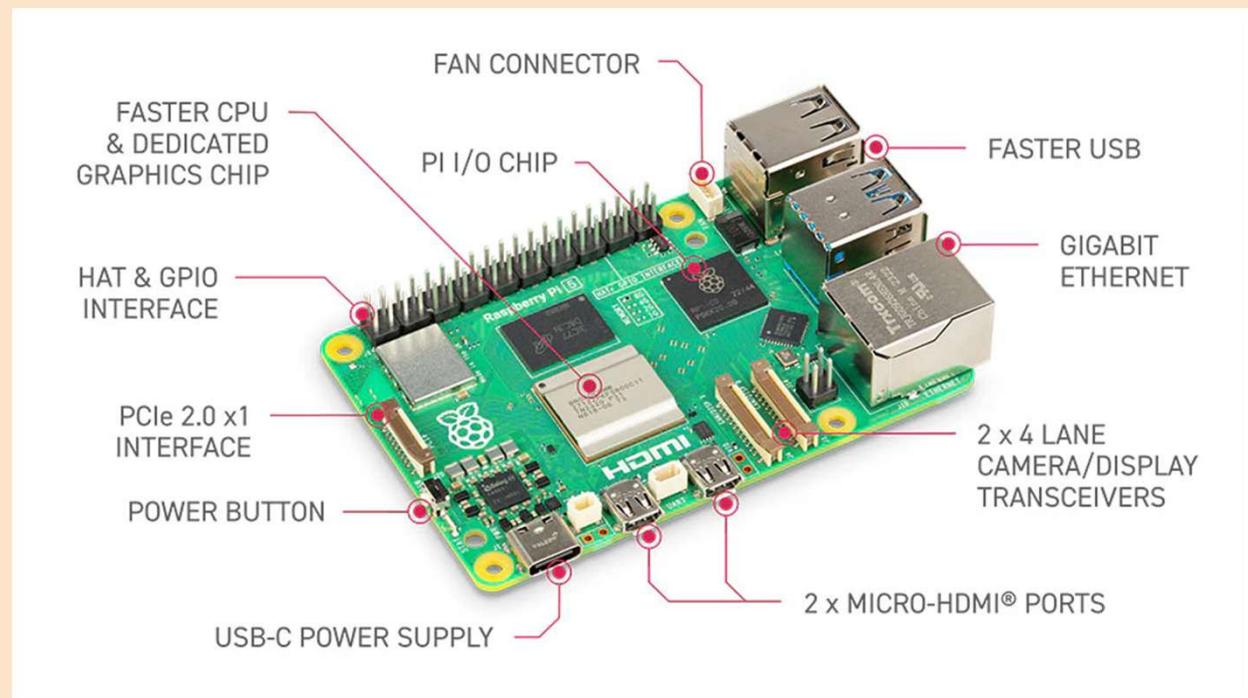


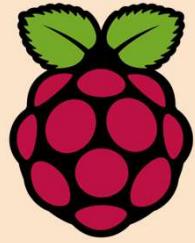
Raspberry Pi?

What is it & Applying electronics

[raspberrypi.com](https://www.raspberrypi.com)

Connectors (for Cameras and so on...)





Raspberry Pi?

What is it & Applying electronics

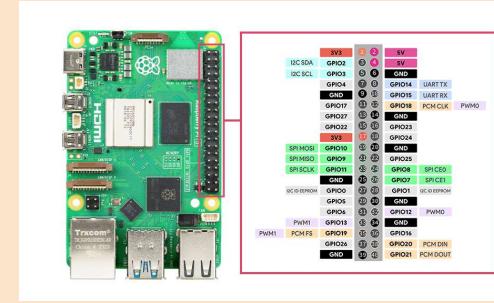
 raspberrypi.com

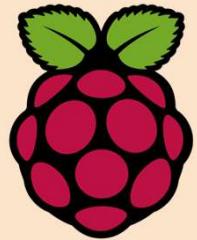
GPIO Pins (General Purpose Input/Output)

What about GPIOs?

- **GPIO Pins** allow the Raspberry Pi to **interact with the physical world**
 - You can connect **LEDs, sensors, motors, buttons, etc.**
 - Ideal for learning about **electronics & physical computing**
 - Combine AI + hardware: e.g. control a light with a model prediction
 - Easy to start with Python libraries like *gpiozero* or *RPi.GPIO*

→ We will apply e.g. the audio-hat here. 





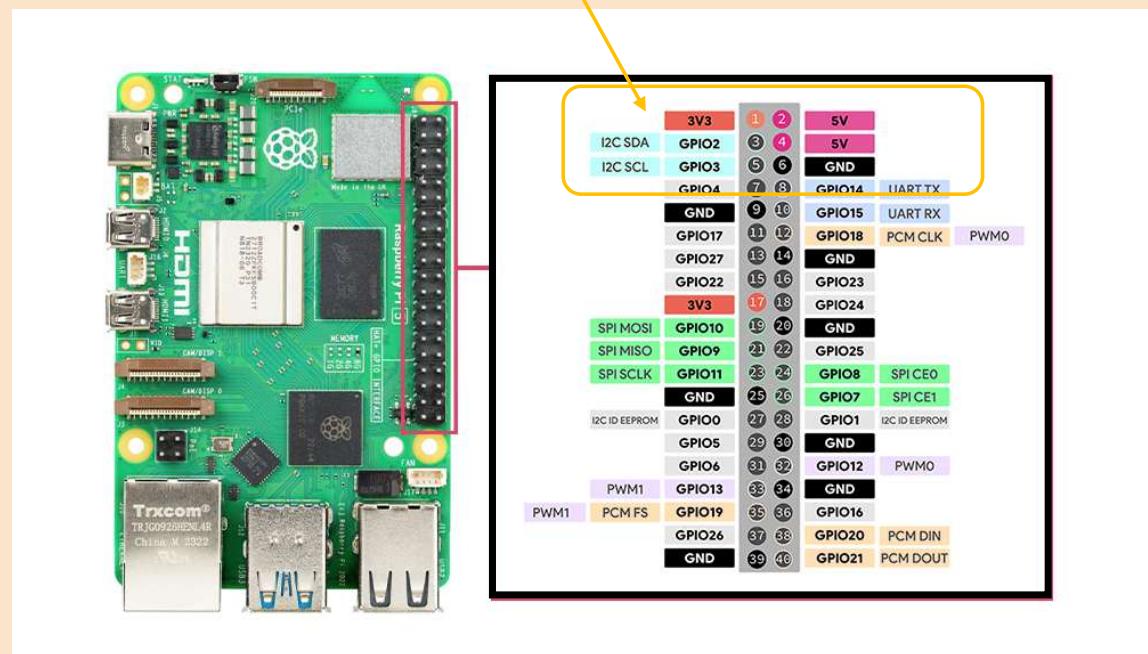
Raspberry Pi?

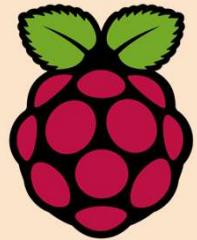
What is it &
Applying electronics

raspberrypi.com

GPIO Pins (*General Purpose Input/Output*)

⊕ Pro Tip: always double check power connectors!



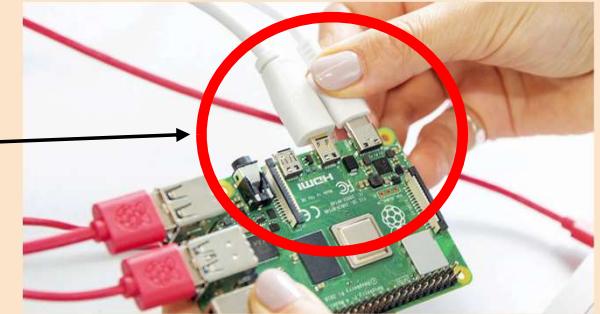


Raspberry Pi?

Let's start it and connect

[raspberrypi.com](https://www.raspberrypi.com)

Plug-in the power cable:
No credentials needed
sudo password: rpi5a,b,c or d



How to connect:
We'll use VNC (Virtual Network Computing) to login remotely to the Pi via the workshop laptops. Check for the icon of RealVNC in the taskbar.

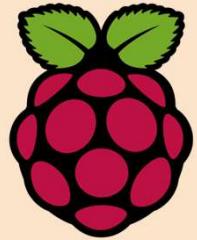
What you need to connect:

- IP-address (see next slide)
- Credentials: rpi5a,b,c or d

Troubleshoot

- *Check if you are in the correct network (ai-spielplatz)*





Raspberry Pi?

Let's start it and connect

IP-Addresses:

rpi5a - inet 192.168.0.136

rpi5b - inet 192.168.0.149

rpi5c - inet 192.168.0.188

rpi5d - inet 192.168.0.113

...Cool! Now we should have remote access to our RPis

Any problems so far? How is the pace?



...speaking Sensors:

Which Sensors do we have?



🛠 ...speaking
Sensors:

Audio Hat



🔧 ...speaking Sensors: Audio Hat

Waveshare WM8960 Audio HAT 🎵 Hi-Fi Stereo Sound Card for Raspberry Pi

🧠 What is it?

- A compact, low-power **audio expansion board** for Raspberry Pi
- Enables **high-quality stereo playback and recording**
- Ideal for **AI voice assistants, audio processing, and audio recognition**

🔧 Key Features

- **WM8960** stereo CODEC with I2S audio interface
- **Dual MEMS microphones** for stereo recording
- **3.5mm headphone jack** and **dual speaker outputs**
- Compatible with Raspberry Pi's 40-pin GPIO header

⚙️ Setup Requirements

- Install drivers provided by Waveshare [[Wiki](#)]
- Utilize **I2C** for control and **I2S** for audio data
- Supports libraries like [pyalsaaudio](#) for using it with python



🛠 ...speaking
Sensors:
RPi Camera



🔧 ...speaking Sensors: RPi Camera

Raspberry Pi Camera Module 3 📸 High-Quality Imaging for AI Projects

🧠 What is it?

- A compact, high-resolution camera designed for Raspberry Pi
- Features a **12MP Sony IMX708 sensor** with **HDR** and **autofocus**
- Ideal for **Computer Vision, machine learning** (data collection / inference), **object detection, robotics & automation** tasks

🔧 Key Features

- **12MP resolution** (4608×2592 pixels)
- Connects via **15-pin MIPI CSI-2 interface**

⚙️ Setup Requirements

- Compatible with all Raspberry Pi
- Utilizes the [libcamera](#) library for image capture and control, usable with [opencv-python](#)



Now... What about AI?

AI Cheatsheet 🧠

Best Practices & Practical Trade-Offs

📌 Core AI Principles

- **Start simple** → Test ideas with basic models first
- **Data > Model** → High-quality data beats fancy algorithms
- **Avoid overfitting** → Your model should generalize, not memorize
- **Evaluate properly** → Always use training **and** test sets
- **Explainability matters** → Know **why** your model makes decisions

- **Keep it reproducible & document it** → Use version control, and logging

Now... What about AI?

Typical Trade-Offs

 Do This	 Instead of This	Why?
Use Pretrained Models	Build from Scratch	Saves time & compute
Clean & Label Data Well	Add More Data Unfiltered	Quality > Quantity
Use Baseline Model First	Jump to Deep Nets	Know your baseline!
Simpler, Interpretable Models	Complex, Black-Box Models	Easier debugging & trust
Open-Source Tools (e.g. scikit-learn, Hugging Face Transformers, pytorch)	Custom from Scratch	Faster development & better support incl. pretrained model support

Pro Tip:

Don't aim for perfection first. **Fail fast, learn faster**, then iterate with better models and cleaner data.

A day... in the life of an AI model

Core AI Concepts

From Data to Inference – How AI Works

1. Data Collection

- Everything starts with **data** – images, text, numbers, sensors...
- The better the data, the better the model

2. Data Preprocessing

- Clean, structure, and prepare the data
- Remove noise, fill missing values, normalize formats

3. Model Training

- The model **learns patterns** from labeled data
- Adjusts **internal parameters** to reduce errors
- Example: "This is a cat" vs. "This is not a cat"

A day in the life of an *AI model*

Core AI Concepts

From Data to Inference – How AI Works

1. Data Collection

- Everything starts with **data** – images, text, numbers, sensors...
- The better the data, the better the model

2. Data Preprocessing

- Clean, structure, and prepare the data
- Remove noise, fill missing values, normalize formats

3. Model Training

- The model **learns patterns** from labeled data
- Adjusts **internal parameters** to reduce errors
- Example: "This is a cat" vs. "This is not a cat"

A day in the life of an *AI model*

Core AI Concepts

From Data to Inference – How AI Works

1. Data Collection

- Everything starts with **data** – images, text, numbers, sensors...
- The better the data, the better the model

2. Data Preprocessing

- Clean, structure, and prepare the data
- Remove noise, fill missing values, normalize formats

3. Model Training

- The model **learns patterns** from labeled data
- Adjusts **internal parameters** to reduce errors
- Example: "This is a cat" vs. "This is not a cat"

A day in the life of an *AI model*

Core AI Concepts

From Data to Inference – How AI Works

4. Model Testing / Validation

- Check performance on **unseen data**
- Helps detect overfitting or underfitting
- Use separate **validation and test sets**

5. Model Tuning

- Adjust settings like learning rate, architecture, etc.
- Called **hyperparameter tuning**

6. Evaluation

- Use metrics like **accuracy, precision, recall, F1-score**
- Helps understand real-world performance

7. Inference

- Final model is deployed to make **predictions** on new data
- Fast and efficient – runs in apps, servers, or on edge devices like Raspberry Pi

**So then,
let's go!**

Learning by
do-ing

1 Ask your Question(s)! Discuss... 🧠

2 You can't break anything, no worries! 🤪

3 Have Fun 🎉



Let's get
practical!



Dream Big...



**Let's discuss
ideas.**