

The background is a vibrant blue with a complex pattern of white and light blue circuit lines, resembling a printed circuit board (PCB). These lines are interspersed with numerous small, light blue circles of varying sizes. In the center of the image is a solid black rectangle with rounded corners. The title 'ROOMTRACR' is centered within this rectangle in a large, white, sans-serif font. Below the title, the names 'JASMIN HUTTUNEN & JONAS JULIN' are centered in a smaller, white, sans-serif font.

ROOMTRACR

JASMIN HUTTUNEN & JONAS JULIN



WHAT IS ROOMTRACR?

- A sensor to calculate room occupancy rate
- Gathers environmental data
- Use the data to calculate % of usage per day
- Why? Health. ↪

Health: The more data to know what particles in the air, pressure, oxidization etc., easier to monitor living conditions.

WHAT WAS NEEDED



How do we gather data → what devices give us what we need?



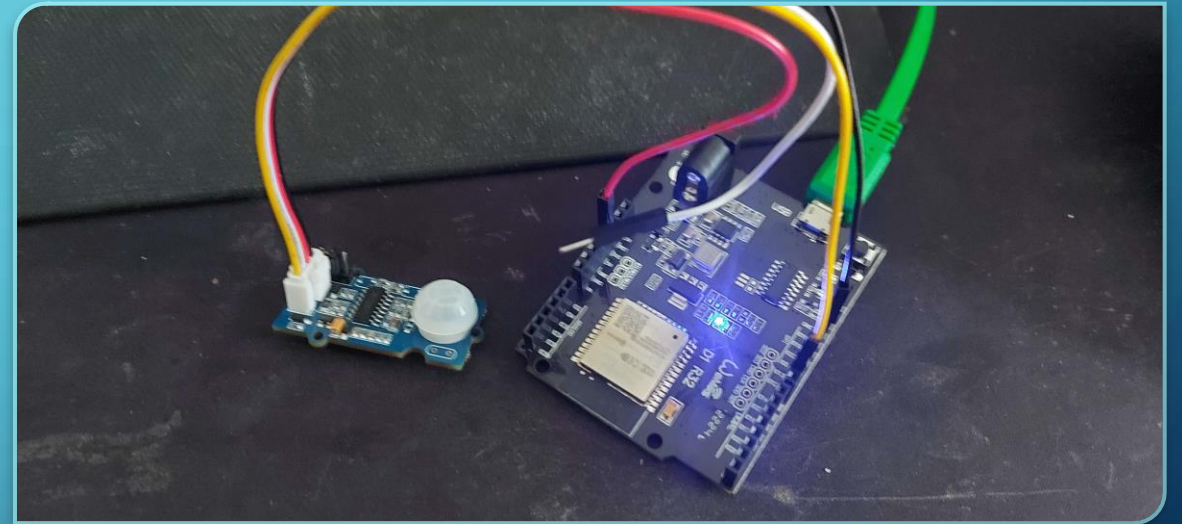
Where do we store the data?



How do we visualize the data and calculate the room occupancy rate?

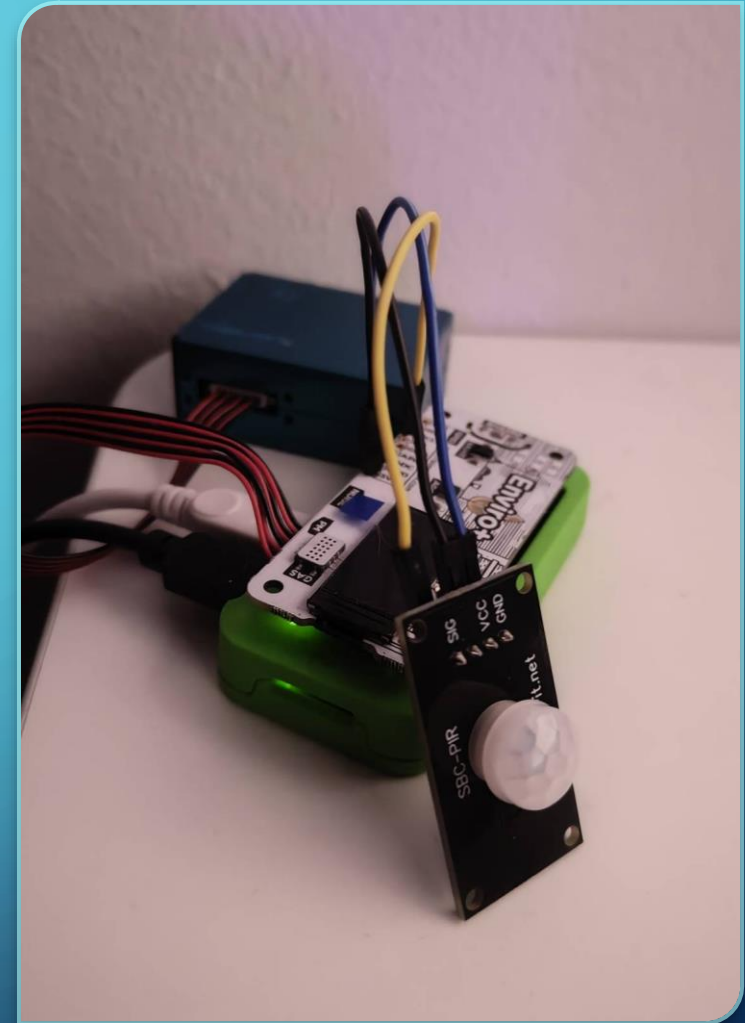
HOW IT WAS ACHIEVED: ESP32 + MOTION SENSOR

- Connect ESP32 with the motion sensor (in this case, grove to jumper cable) → connect via USB
- Set up Arduino IDE
 - Make Arduino code to set up motion sensor
- Set up a database, enter a table with relevant columns
- Then either:
 1. Connect Arduino to a Raspberry Pi for easier connection and data sending
 2. Connect the Arduino to WiFi (if applicable), proceed to send data via HTTP POSTs every 5-10 seconds



HOW IT WAS ACHIEVED 2: RASPBERRY PI + ENVIRO

- Used components:
 - Raspberry Pi Zero W
 - Pimoroni Enviro+
 - PIR motion sensor
- Set up → Connect to WiFi → Install Enviro+ library → connect PIR motion sensor
- Create a table on the database with relevant columns
- Use of a python script to read sensors and send the sensor-data to database in 10 sec. intervals



HOW IT WAS ACHIEVED 3: DATABASE



We needed a database to store and gather sensor data



Options were to do local or cloud → we chose cloud.

NB: Can be done locally!



We used CSC.fi's free student resources:

- Service: Pukki (DBaaS)
- Provides PostgreSQL / MariaDB
- Daily backups (retained for 90 days)
- Built-in firewall management
- Easy setup & connection



Data sent directly to the DB and DB connected to Grafana.

We could then do SQL queries for data visualization

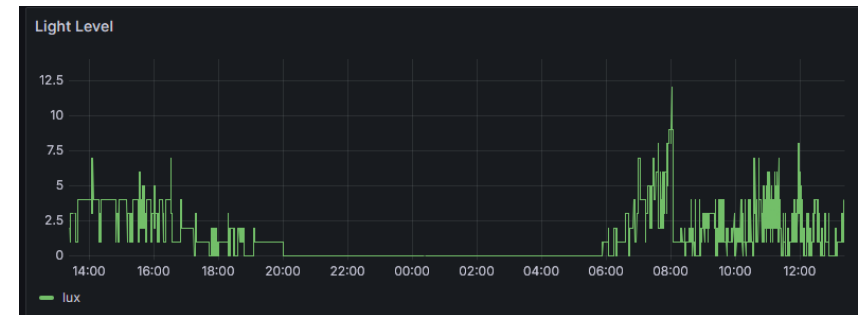
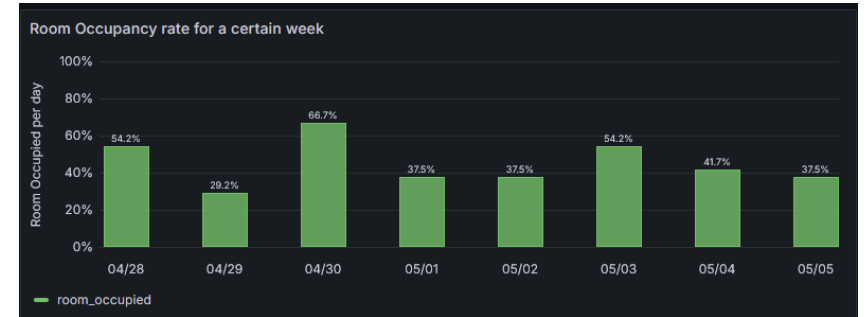
The background of the slide features a blurred image of server racks in a data center, with blue and yellow lights visible. Overlaid on the left side is a white line-art graphic of a circuit board with various nodes and connecting lines.

NOTE ABOUT THE DATABASE

- As previously mentioned, we used CSC.fi for the project
- However, the project was marked to end on 09.05.2025 → no access to the service or database after that date (and could not be extended as it was a student project)
- Jasmin set up a new PostgreSQL database on a private Hetzner server
 - Database migrated from a dump of the CSC database
 - Grafana reconfigured to connect to the new database
 - The Raspberry Pi's .env file was updated with the new database connection details (IP address, username, database name, password)
 - Data collection and visualization continued seamlessly

THE DATA VISUALIZED

- We used only the Raspberry Pi's sensor data for visualization
→ Easier as it gathered everything from one spot
- We used Grafana for visualization
- This was needed for hard evidence of the sensors working
- This was also needed for the calculations for room occupancy rates
 - Several data points were used for the sake of calculation, using motion sensor data as a base to say the room is being occupied
- The Grafana is hosted on Jasmin's Hetzner [server](#)(requires user+pw+access)



POSSIBLE FURTHER DEVELOPMENT

- The sensor could possibly be built cheaper
- A design of a wall-mounted sensor with LED's and audible alerts for bad air quality
- A possible application to show the data, visualizations and alerts

FULL DOCS - GITHUB

<https://github.com/J-Huttunen/RoomTracr>

Jasmin's Github page with documentation on the setup, with Jonas' documentation on the ESP32 usage.

NB: It's in finnish.

An abstract graphic on the left side of the slide, consisting of white lines and circles on a blue background, resembling a circuit board or a stylized tree structure.

THANK YOU

A single vertical white line located on the right side of the slide.