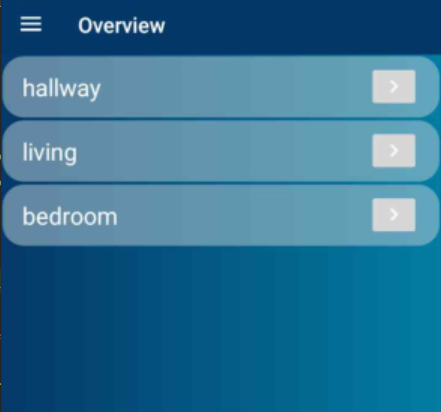
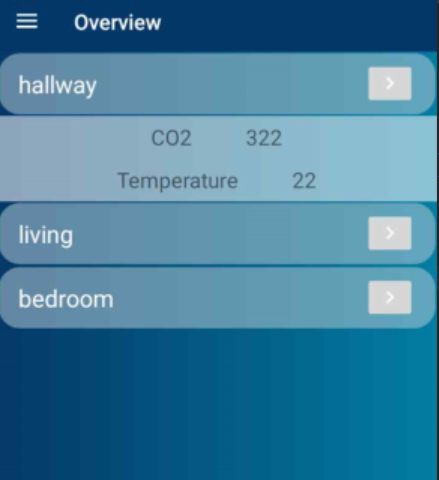
Expandable recycler view

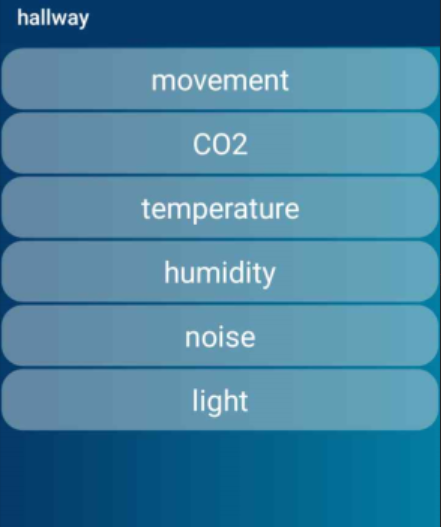
In the Android application, we used an open source library for the recycler view part. The purpose of custom recycler view adapters is to expand and collapse groups with support for multiple view types.

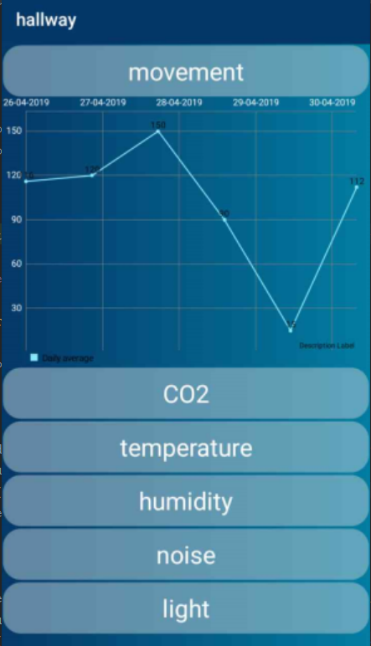


In our case, we used this library for the Sensors view. By expanding and collapsing the sensor view holder you are able to see more or less information about the sensor data. In the second sensor content view, by pressing on the sensor name you can see a data chart. Both views use this expendable recycler view library.



🡪





🡪

To make it work, two different view holders were created for each view. One ChildViewHolder and one GroupViewHolder. The GroupViewHolder contained the child one, and by pressing on the main view, you would expand it and be able to see the contents of the ChildViewHolder.

Retrofit 2

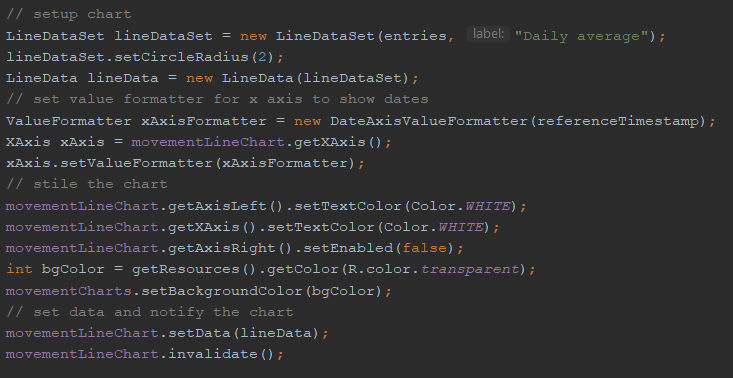
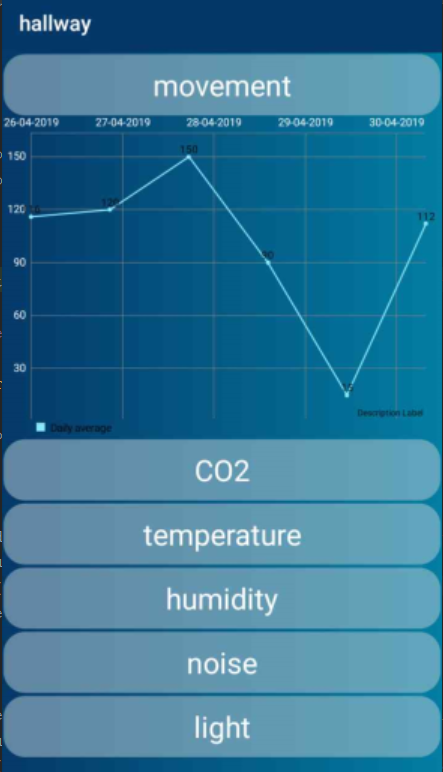
Retrofit is an HTTP client for Android and Java. What it does is that it simplifies HTTP communications by turning remote APIs into type-safe interfaces.

\*Michaela already did the documentation for retrofit.

MPAndroid Charts

To be able to use charts to display data in the application we used the MPAndroid Charts library developed my Philipp Jahoda. The way that this library works is that it can show data in the form of a chart of your choosing, from line charts to pie charts, or candlestick charts. The charts also support scaling, dragging and different animations.

We used this library for displaying the data received from the sensors in a more user-friendly way.



The above code represents the process from the initialization of the charts then getting the actual data from the sensor with a timestamp and then setting up the chart and styling it to be easy to read by the user.