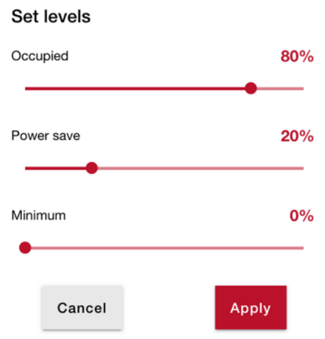
Helvar Front-end Developer Home Assignment

Task

Your task is to implement a component for setting light levels for an intelligent luminaire. The luminaire has a PIR sensor, so it can detect movement. Please use the following design as a reference. The implementation does not need to be pixel-perfect but similar looking.

Light level definitions

- **Occupied**: The sensor has seen movement in the last 5 minutes.
- **Power save:** The sensor has not seen the movement in the last 5 minutes.
- **Minimum:** The sensor has not seen movement in the last 10 minutes.



Requirements

- It's enough if the UI looks good with mobile devices, no need to make it responsive.
- The component has 3 sliders, one for each light level.
    - The user can see values updating live when dragging a slider.
    - The levels can have the following values [0%, 1%, 5%, 10%, 15%, …, 100%]
        - Initial values:
            - Occupied: 80%
            - Power save: 20%
            - Minimum: 0%
    - **Restrictions**
        - "Occupied level" must always be equal or greater than "Power Save" and "Minimum"
        - "Power save" must always be equal or greater than "Minimum"

        - The value of the other sliders should change automatically so that the conditions above always apply (check the attached *levels-sliders-clip.mov*)
- When pressing “Apply”, the component emits the levels which have been set.
- When pressing “Cancel”, the component emits a value to indicate that the user does not want to update the luminaire light levels.
- The code needs to have automated tests to ensure it works.

Instructions

- You need to use either Angular or React for the implementation. You can choose any tool you prefer for writing the automated tests.
- You can use any dependencies which can be used in a commercial product (license should be MIT, BSD, or similar).
- Please upload the source code to Github or similar so that we can review it.

Grading (max 5 points)

- Good code and development practises (2 points)
- UI looks similar (1 point)
- Functionality correct (1 point), and automated tests written (1 point)