<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Smart Home Automation Layout - Square Rooms</title>

<style>

body {

font-family: "Segoe UI", Arial, sans-serif;

background: #e9eff4;

display: flex;

flex-direction: column;

align-items: center;

padding: 20px;

}

h1 {

color: #333;

margin-bottom: 25px;

}

/\* Grid layout for square rooms \*/

.house {

display: grid;

grid-template-areas:

"hall bedroom1"

"bedroom2 kitchen";

grid-template-columns: 1fr 1fr;

grid-template-rows: 1fr 1fr;

gap: 12px;

width: 95%;

max-width: 900px;

height: 900px; /\* square overall \*/

position: relative;

}

/\* Room styling \*/

.room {

border: 4px solid #b0bec5;

border-radius: 10px;

display: flex;

flex-direction: column;

align-items: center;

flex-wrap: wrap;

padding: 20px;

position: relative;

overflow: auto;

}

.room h2 {

position: absolute;

top: 5px;

left: 10px;

font-size: 16px;

color: #333;

font-weight: 600;

}

/\* Room colors \*/

.hall { background: #fff9c4; grid-area: hall; }

.bedroom1 { background: #bbdefb; grid-area: bedroom1; }

.bedroom2 { background: #c8e6c9; grid-area: bedroom2; }

.kitchen { background: #ffe0b2; grid-area: kitchen; }

/\* Devices \*/

.device {

display: flex;

flex-direction: column;

align-items: center;

margin: 15px;

width: 100px;

cursor: default;

text-align: center;

position: relative;

}

.icon {

font-size: 36px;

filter: brightness(0.7);

transition: 0.4s ease;

position: relative;

}

.label {

font-size: 14px;

margin-top: 6px;

}

.switch {

appearance: none;

width: 50px;

height: 24px;

background: #ccc;

border-radius: 24px;

position: relative;

margin-top: 5px;

cursor: pointer;

transition: 0.3s;

}

.switch::before {

content: "";

position: absolute;

top: 2px;

left: 2px;

width: 20px;

height: 20px;

border-radius: 50%;

background: white;

transition: 0.3s;

}

.switch:checked {

background: #4caf50;

}

.switch:checked::before {

left: 28px;

}

/\* ON state animations \*/

.on .icon { filter: brightness(1.5); transform: scale(1.1); text-shadow: 0 0 15px gold; }

.on .fan { animation: spin 1s linear infinite; }

.on .light { color: #ffd600; text-shadow: 0 0 20px yellow; }

.on .tv { color: #00bcd4; text-shadow: 0 0 25px cyan; }

.on .ac::after { content: "💨"; position: absolute; top: 35px; font-size: 20px; animation: air 1s infinite alternate; }

.on .fridge { color: #00b0ff; text-shadow: 0 0 10px #00e5ff; }

.on .smoke { color: red; animation: pulse 1s infinite; }

.on .exhaust { animation: spin 1s linear infinite; }

@keyframes spin { from { transform: rotate(0deg); } to { transform: rotate(360deg); } }

@keyframes air { 0% { opacity: 0.2; transform: translateY(0px);} 100% { opacity: 1; transform: translateY(-10px); } }

@keyframes pulse { 0% { opacity: 0.6; transform: scale(1);} 50% { opacity: 1; transform: scale(1.2);} 100% { opacity: 0.6; transform: scale(1);} }

/\* Sensor Box \*/

.sensor {

margin-top: 25px;

background: white;

padding: 15px 25px;

border-radius: 10px;

box-shadow: 0 2px 8px rgba(0,0,0,0.1);

max-width: 400px;

text-align: center;

}

.sensor p { font-size: 18px; margin: 5px 0; }

</style>

</head>

<body>

<h1>🏠 Smart Home Automation - Square Rooms</h1>

<div class="house">

<!-- Hall -->

<div class="room hall">

<h2>Hall</h2>

<div class="device" id="hall\_tv">

<div class="icon tv">📺</div>

<div class="label">TV</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="hall\_fan">

<div class="icon fan">🌀</div>

<div class="label">Fan</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="hall\_light">

<div class="icon light">💡</div>

<div class="label">Lamp</div>

<input type="checkbox" class="switch">

</div>

</div>

<!-- Bedroom 1 -->

<div class="room bedroom1">

<h2>Bedroom 1</h2>

<div class="device" id="bedroom1\_ac">

<div class="icon ac">❄️</div>

<div class="label">AC</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="bedroom1\_fan">

<div class="icon fan">🌀</div>

<div class="label">Fan</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="bedroom1\_light">

<div class="icon light">💡</div>

<div class="label">Lamp</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="bedroom1\_tv">

<div class="icon tv">📺</div>

<div class="label">TV</div>

<input type="checkbox" class="switch">

</div>

</div>

<!-- Bedroom 2 -->

<div class="room bedroom2">

<h2>Bedroom 2</h2>

<div class="device" id="bedroom2\_ac">

<div class="icon ac">❄️</div>

<div class="label">AC</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="bedroom2\_fan">

<div class="icon fan">🌀</div>

<div class="label">Fan</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="bedroom2\_light">

<div class="icon light">💡</div>

<div class="label">Lamp</div>

<input type="checkbox" class="switch">

</div>

</div>

<!-- Kitchen -->

<div class="room kitchen">

<h2>Kitchen</h2>

<div class="device" id="kitchen\_light">

<div class="icon light">💡</div>

<div class="label">Lamp</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="kitchen\_smoke">

<div class="icon smoke">🔥</div>

<div class="label">Smoke</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="kitchen\_exhaust">

<div class="icon exhaust">🌀</div>

<div class="label">Exhaust</div>

<input type="checkbox" class="switch">

</div>

<div class="device" id="kitchen\_fridge">

<div class="icon fridge">🧊</div>

<div class="label">Fridge</div>

<input type="checkbox" class="switch">

</div>

</div>

</div>

<div class="sensor">

<h2>🌡️ Bedroom 1 Sensor Data</h2>

<p id="temp">Temperature: -- °C</p>

<p id="humidity">Humidity: -- %</p>

</div>

<script>

document.querySelectorAll('.device').forEach(device => {

const toggle = device.querySelector('.switch');

toggle.addEventListener('change', () => {

device.classList.toggle('on', toggle.checked);

});

});

// Bedroom 1 sensor simulation

function updateBedroomSensor() {

const temp = (22 + Math.random() \* 6).toFixed(1);

const humidity = (45 + Math.random() \* 15).toFixed(1);

document.getElementById('temp').textContent = `Temperature: ${temp} °C`;

document.getElementById('humidity').textContent = `Humidity: ${humidity} %`;

}

setInterval(updateBedroomSensor, 3000);

updateBedroomSensor();

</script>

</body>

</html>