# Scenario

You are currently working as a junior software developer at CUBE Music Pty Ltd, a Perth-based company that creates specialised music player software. They have a variety of different solutions, which include streaming platforms (e.g., Spotify, TIDAL, etc.) and interfaces with a range of different HiFi equipment (e.g., DAC’s).

CUBE Music’s software ecosystem is called (no surprise) CUBE.

The company is looking into adding features to CUBE that will allow smart multi-room music playback, including some form of “follow me”, which will use various IoT sensors.

Table of Contents

[Scenario 1](#_Toc79318818)

[General Information 3](#_Toc79318819)

[Follow Me 3](#_Toc79318820)

[Prototype Outline 4](#_Toc79318821)

[Monitoring 4](#_Toc79318822)

[Software 4](#_Toc79318823)

[Sensors and Actuators 4](#_Toc79318824)

[Additional Optional Features 5](#_Toc79318825)

# General Information

The director of the company, JT, is always looking for innovative ways to expand the business. She is very interested in IoT and what it might add to the company’s music software portfolio, which is exactly why you have been hired.

## Follow Me

The company is currently investigating the possibility to add a “follow me” option to the music software. This feature would use various IoT sensors to determine whether someone is in a room and what music should be played there.

The research is still in its infancy and the company needs to figure out what sensors to use and how they will interact with the CUBE ecosystem. Despite the company operating at the high-end of the market, they have decided that any prototype will need to be low-cost and built from readily available (COTS[[1]](#footnote-2)) components, like, e.g., the Raspberry Pi, ESP32.

The initial prototype may use some of the following sensors: door switch (open/close – reed relay), motion sensors (e.g., PIR), presence sensors (they don’t detect motion, but someone’s presence – very convenient to detect people who are not moving), IR gates, but also temperature, etc. At some point, cameras may be considered. Any sensors that a device has “out of the box” will also be incorporated, e.g., temperature, ambient light, etc.

Since ‘Follow Me’ will be incorporated into the CUBE multi-room system, the system should be able to assign different sensors to different areas (e.g., rooms, garden, etc.) in a house.

Later versions will also leverage the presence of, for example, mobile phones and smart watches. Using these devices, ‘Follow Me’ can be truly personalised, so that different people can listen to different music (when in different rooms). This also means that ‘Follow Me’ needs to be extensible.

Ultimately, any devices that will be deployed as part of ‘Follow Me’ need to be able to connect to a network. This can be either a wired connection (e.g., Ethernet) or wireless (WiFi, …) This is outside the scope of this prototype, though.

‘Follow Me’ should also be “backward compatible”. This means that the company cannot assume that sensors will be installed where wires are available. In some cases, low-power battery-operated wireless sensors may need to be installed.

# Prototype Outline

To assist, and to demonstrate your abilities, the company have set a task that requires you to develop a proof-of-concept prototype that will form the basis for further research into ‘Follow Me’.

The prototype will be self-contained using a Raspberry Pi and SenseHAT, and/or other equipment as required (e.g., GPIO-based sensors, I2C or SPI-based sensors).

The prototype is to have the following features:

* Monitoring of Presence in a Room using Various Sensors SENSEHAT
* Monitoring of Prototype Hardware & Other Factors (device health) SENSEHAT
* Storing data into and retrieving data from a database ADAFRUIT
* Displaying a summary dashboard and detailed information ADAFRUIT
  + for at least one room
  + Using at least two different sensors

## Monitoring

The prototype is to have a minimum set of features that include:

* Monitoring of Presence including:
  + GPIO-based sensors (e.g., door switches, PIR, …) SENSEHAT joystick
  + Simulated sensors
* Monitoring of Prototype Hardware to determine device health, including:
  + CPU Load, SENSEHAT SENSORS
  + Device Temperature, and
  + Storage Space.

## Software

The prototype solution will need to provide some essential features. The features that will need to be considered and software solutions created include:

* Data storage within the prototype hardware LOCAL STORAGE FILE
* Regular “upload” to centralised data storage system ADAFRUIT
* Displaying a dashboard of readings as numerical values ADAFRUIT
* Historical data display for readings using appropriate charts. ADAFRUIT
* Historical for a prescribed period of time (for example the last 5 minutes). ADAFRUIT

## Sensors and Actuators

The lists before show a number of possible sensors and actuators that may be useful in solving this scenario. The solution may use some, none or all of these and even others that are not listed.

|  |  |
| --- | --- |
| Sensors  * Camera * Bluetooth/wireless * Temperature sensor * Light level sensor * IR light gates * Reed switches * PIR sensor * Ultrasonic movement sensor | Actuators  * LED Display * Bluetooth/wireless * Power switching (relays) * Buzzer |

## Additional Optional Features

Additional features may be added to the system if desired but are not part of the main task and will not affect your job.

These additional features may include, but are not limited to:

* Power Status
* Battery Level monitoring
* Light levels in rooms
* Activity in selected areas
* Data uploaded to central data storage at regular intervals
* A web page showing current readings (automatically updating)
* A web page showing results/readings for the required reporting period
* Automated light adjustment based on activity in area
* Heat map of use of areas/rooms in the house (i.e., high-traffic areas vs low-traffic areas)
* Reports showing use of equipment and consumables based on periods that may include 24 hours, 7 days, 28 days, calendar monthly and yearly.

1. *COTS=Commercial-of-the-shelf* [↑](#footnote-ref-2)