Smart IoT Extractor fan

Ayham Treter
Computer Engineering Department, ISIK University

1 Introduction to the Problem

All houses nowadays have extractor fans, specifically in bathrooms and kitchens, yet they are currently only activated by manual switches. Despite the widespread use of these fans, there has been little advancements in terms of automation and modularity. This has led to the creation of a gap in the market for either high-end smart ones or simpler cost-effective. This problem gave us the idea to create a modular simple system to automate them.

2 Your Method to Solve the Problem

Our method to solve the problem is to create an IoT device that can connect to any existing extractor fan and make it smart by using an ESP8266 chip to connect to the internet and process various sensors data to measure the temperature and humidity of the room along with way to connect any fan via wires.

In terms of design decisions, we chose to use an ESP8266 chip as it is a low-cost and widely available microcontroller that can easily connect to the internet and process sensor data. We also chose to include sensors for measuring temperature and humidity as these are key factors in determining when an extractor fan should be activated, Further,as these sensors may sometime give incorrect reading, we have included some software pre-processing safeguards for negative readings and also temperature sensor voltage reading. As for Post-processing, these data will be used to make localized decisions about activating the fan and store the data on the cloud for graphs. Additionally,we included NPN transistor to support fan integration up to 12V.

To justify the choices made, the use of ESP8266 chip as the microcontroller is a cost-effective and widely available option that allows for easy internet connectivity and processing of sensor data. The inclusion of sensors for measuring temperature and humidity is necessary as these are key factors in determining when an extractor fan should be activated.

The software pre-processing safeguards for negative readings and temperature sensor voltage reading help to ensure that the data received by the cloud server is accurate and reliable, which is important for the proper functioning of the system. This will also help the system to be more robust and less prone to errors.

The use of post-processing to make localized decisions and storing the data on the cloud to allows for efficient and convenient control and view of system anywhere. The localized decision-making allows for more energy efficiency and automation which is our aim for designing this system. Storing the data on the cloud also allows for real-time viewing of the data and can help the user identify trends in the data.

In addition, the inclusion of an NPN transistor to support fan integration up to 12V is a practical solution that allows the system to control a wide range of extractor fans and ensures compatibility with a variety of different fans. This will increase the flexibility and adaptability of the system.

2.1 Software Design

We have used DHT library for read data from the humidity sensor, we also used Blynk cloud to integrate our hardware with the cloud and connect it to the internet easily.

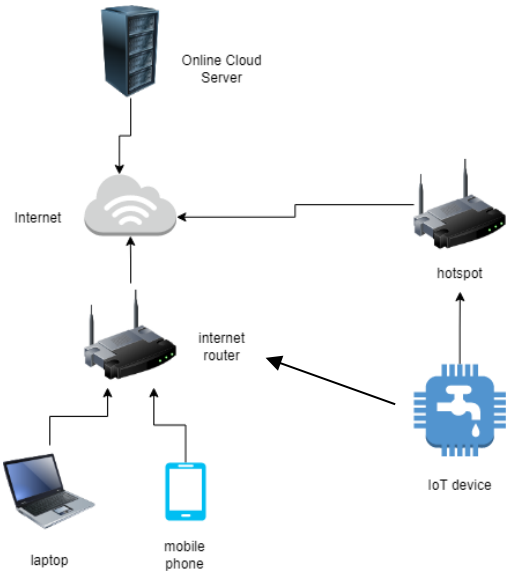

Figure 1: The proposed system

3 Results and Analysis

To evaluate the performance of the proposed IoT fan system, several tests were conducted under different conditions. Room Temp and humidity were placed in center of the bathroom, and Water temp was placed near the pipes (touching it) The results are presented in the table below

*Override includes both physical and software buttons
*Thresholds: Humidity: 60%, Room temp: 30°C, Water temp: 31°C

Test	**Humidity**	**Room Temperature**	**Water Temperature**	**Automatic Activision**	**Override**
Test 1	61%	25°C	26°C	Successful	Successful
Test 2	65%	31°C	26°C	Successful	Successful
Test 3	70%	32°C	33°C	Successful	Successful
Test 4	50%	31°C	26°C	Successful	Successful
Test 5	55%	29°C	31°C	Successful	Successful
Test 6	50%	25°C	26°C	Fail (This is successful)	Successful

*Highlights shows that system was able to activate on a single condition only

As shown in the table above, the proposed system was able to successfully activate the fan automatically when the temperature and humidity thresholds in the 3 conditions were reached in all test cases. Additionally, the manual override feature was also successful in all test cases, allowing the user to manually activate or deactivate the fan as needed. And Test 6 shows that the system turns off when threshold are not met but override still makes it spin.

The temperature and humidity data collected by the system during the tests was analyzed to identify some patterns. The data showed that the system was able to measure the temperature and humidity of the the room.

A few challenges were encountered during the testing process, including some issues with the fan integration, however, these were resolved by using an NPN transistor to support fan integration up to 12V And Settings the threshold was also a challenge to obtain the optimal results for our test room

4 Conclusions and Future

The proposed IoT fan system has the potential to be a cost-effective and practical solution for controlling extractor fans in homes. The system was able to successfully meet the objectives of the project during initial testing, and the results indicate that the system is able to provide a convenient and efficient way to control extractor fans in bathrooms and kitchens.

The system was able to automatically activate the fan when certain thresholds were reached, and the manual override feature provided additional flexibility for the user in both physical and software layer, and it also means the fan should also function in case of the system going down by using the physical button. The data analysis also showed that the system was able to measure the temperature and humidity of the room, and that the thresholds set for automatic activation were appropriate for the environment in which the tests were conducted.

However, it's important to note that further testing and validation is needed in different environments and under different usage scenarios to ensure that the system is working correctly and efficiently in different conditions. Additionally, the use of open-source software and firmware will allow the system to be customized and improved as needed, which will make it more adaptable to different environments and usage scenarios.

It is also important to note that the system is modular with its components, which makes it easy to upgrade or replace any individual component as needed. This will increase the lifespan of the system and make it more cost-effective in the long run. Additionally, the modular design allows for easy integration with other smart home systems

In terms of future work, further testing and validation in different environments and under different usage scenarios is needed to ensure the system's performance and robustness.

It would be useful to explore the integration of additional sensors such as air quality sensors, to enhance the system's ability to detect and respond to changes in air quality.

integration of machine learning algorithms to improve the system's decision-making capabilities and set automatic threshold rather than fixed values, in order to optimize the system's performance

Other works could also be Integration with smart home systems such as Samsung Smart-things rather than standalone application to enable seamless integration with other smart devices in the home