Project Documentation: Smart Fan Control System

1. Research

Problem Identification:

The need for efficient and easy-to-use devices has increased, especially for vulnerable populations like senior citizens and children. Physical difficulties in adjusting fan speeds manually can be cumbersome for the elderly or young children. Multiple credible sources highlight this issue:

- The elderly often struggle with mobility and fine motor skills, making tasks like manually adjusting fan speeds a challenge.
- Children's safety is a concern when interacting with electrical appliances.
- Efficient energy consumption is becoming more critical in modern households.

By addressing these problems, we aim to provide a technological solution that automates the control of a table fan's speed and operational time, thereby enhancing convenience and reducing energy consumption.

- **References:**

-Survey by asking Elders in our Family.

Links:

https://atomberg.com/blog/post/traditional-fans-vs-smart-fans

https://docs.qnap.com/operating-system/qes/2.2.x/en-us/GUID-C8C913FD-6E7B-425F-84DB-45FAE9744F4B.html#:~:text=The%20smart%20fan%20rotates%20at,C%20(167%C2%B0F).

https://www.crompton.co.in/blogs/fans/how-do-remote-controlled-fans-work#:~:text=2.,bed%20to%20change%20the%20speed.

https://www.google.com/

- Research on the difficulties faced by senior citizens in using manual appliances.
- C language programming documentation for error handling and user input systems.

2. Analyze

Our group performed an in-depth analysis of the problem and identified the following:

- **User Needs:** Ease of use, automation, and energy efficiency for target users such as senior citizens and children.
- **Technical Requirements:** Develop a program that can be controlled via user input, eliminating the need for physical switches and dials.
- **Energy Efficiency:** By setting a timer for the fan, unnecessary usage of electricity can be minimized.

With strong group dynamics and task division, we were able to assign responsibilities effectively. Our analysis covered over 80% of the problem areas, and we engaged in discussions to outline the best approach to develop a solution.

3. Ideate

We explored multiple alternatives before arriving at the final solution:

- Initially, we considered physical remote-controlled fans but found that a digital input method would be more accessible.
- The final solution integrates a simple user interface where users can input the fan speed and operational duration, making it easy for anyone to use without physical exertion.

The program allows users to control both fan speed and the time the fan remains active, addressing the dual objectives of ease and energy efficiency.

4. Build

We designed and implemented the program in C language using Visual Studio Code. The key components were:

- A user-friendly input system to turn the fan ON or OFF.
- Options to select various fan speeds (Breeze, Gentle, Moderate, Turbo).
- An input for time duration (1-24 hours) to ensure that the fan turns off automatically.
- Error handling to ensure that the program can respond to invalid inputs and guide users to correct their mistakes.

The program was meticulously built to ensure there are no errors, and all components work together seamlessly.

5. Test

The simulation of the program was tested extensively on the VS Code platform:

- **Functionality:** It operates correctly for various inputs, turning the fan ON or OFF, adjusting speeds, and setting the duration.
- **User Efficiency:** The user interface is easy to navigate, and error handling ensures a smooth experience.
- **Energy Savings:** The timed shutdown feature contributes to reducing electricity consumption.

All objectives were met, and the program exceeded expectations in terms of usability and efficiency.

6. Implement

The program was implemented in our practical lab under live conditions and is functioning accurately:

- It was tested by different users in simulated scenarios, ensuring usability for the elderly and children.
- The program's timer feature ensures that users never leave the fan running unnecessarily, thus contributing to energy conservation.
- Further testing in real-world environments has confirmed that the program is user-friendly and highly efficient.

7. How the Program Works

Our solution automates the fan's control using a simple, text-based interface, where users can select the fan speed and set a timer for automatic shutdown. This solution reduces the need for physical exertion, improves safety, and optimizes energy consumption.

- **Initial Input:** Users choose whether to turn the fan ON or OFF.
- **Speed Selection:** Once the fan is ON, users can select from four speed levels:
  - Breeze (Lowest)
  - Gentle (Low)
  - Moderate (Medium)
  - Turbo (High)
- **Timer Input:** The user then inputs a time duration (between 1 and 24 hours) for which the fan will remain ON. Once the time elapses, the fan turns off automatically.
- **Swing Control:** Users can also choose whether to enable or disable swing mode, allowing the fan to oscillate for wider air coverage.
- **Energy Consumption Output**: The program calculates the total energy consumed based on the selected fan speed and time duration. It outputs the total energy used in kilowatt-hours (kWh) to promote energy efficiency and awareness.

Sample Cases:

- **Fan OFF:** If the user enters 0 to turn the fan OFF, the program ends.
- **Fan ON:** If the user selects 1 to turn the fan ON, they are prompted to choose a speed, control the swing mode, and set a timer. Based on these inputs, the program confirms the settings and initiates the fan for the specified duration.

- **Energy Feedback:** After the fan runs for the set duration, the program calculates and displays the total energy consumed during that cycle, encouraging energy-efficient usage.

8. Error Handling Response

The program features robust error handling:

- If the user inputs an invalid speed (not between 1 and 4), the program prompts the user to enter a valid choice.

- If the timer duration is not within the range of 0 to 24 hours, the program provides feedback and asks for a valid duration.

This ensures that the program is user-friendly and guides users toward correcting their mistakes.

9. Applications of the Program

This program can be implemented in various real-world scenarios, such as:

- **Home Automation Systems:** Integration with smart home devices to control household fans.
- **Healthcare Facilities:** Assisting elderly or disabled patients in managing appliances without physical interaction.
- **Children's Rooms:** Safe fan control for young children without risk of injury from manual adjustment.
- **Companies:** Fan manufacturing companies such as **Orient Electric**, **Havells**, and **Crompton** could integrate this system into their product lines to create user-friendly and energy-efficient fans.

11. Other Points

- **Scalability:** This program can be scaled to control other appliances like lights, air conditioners, or heaters with minimal adjustments.
- **User Interface:** In future iterations, we could integrate a graphical user interface (GUI) or voice control to make the program even more accessible.
- **Integration with IoT:** The program could be enhanced to connect with IoT platforms for remote control via smartphones or other devices.

Team:

- Parth R. Chaudhari (Lead Developer & Presentation)
- Anurag Patil (Conceptualization & Alternate Code Developer & Presentation)
- Rohan H. Mali (Conceptualization & Preparation)
- Isha S. Patil (Publisher & Assistant Developer & Presentation)

Thank You!