AME 494 A6

Problem/Task

The primary goal is to create an IoT system that provides real-time temperature and humidity monitoring for a room and then cast it to your phone. The extended vision is to evolve this system into a fashion app that suggests outfits based on the weather conditions.

- In my head, this application could be used as a fun way for people who are just getting into fashion to better understand and go through their days while still dressing simplistically and well.
- However, at a base implementation it will simply read the temperature and humidity and cast it to your phone

Components of the System

- TTGO T-Watch: This is a wearable device equipped with sensors to measure temperature and humidity. It may also have Bluetooth capabilities for data transmission.
- Computer: Acts as a gateway and intermediary between the T-Watch and the phone. It runs a program to receive data from the T-Watch and forwards it to the mobile app.
- Phone: The mobile app on the phone receives, processes, and displays temperature and humidity data in a user-friendly interface.
- Communication Protocol: Suitable protocol for communication between the T-Watch, computer, and phone. MQTT or Bluetooth Low Energy (BLE) are potential options.
- Mobile App (Optional) : Develop a cross-platform mobile application using advanced framework

Interaction between Components

- The T-Watch periodically measures temperature and humidity and sends this data to the computer via a communication protocol (e.g., MQTT or BLE).
- The computer processes the data and forwards it to the mobile app.
- The mobile app updates the UI with real-time temperature and humidity readings.

Trade-offs in Design

- Real-time vs. Power Consumption: Implement strategies to balance the need for real-time updates with the impact on the T-Watch's battery life, such as adjusting the frequency of data transmission.
- Complexity vs. Simplicity: Consider user experience when designing the mobile app, ensuring it provides valuable information without overwhelming the user with complexity.
- Data Accuracy vs. Power Consumption:
    a. *Tradeoff:* Increasing the frequency of data readings on the T-Watch enhances data accuracy but may significantly impact power consumption.
    b. Consider balancing the need for precise real-time data with the T-Watch's battery life. Implement adjustable sampling intervals or intelligent algorithms to optimize power usage based on the significance of changes in temperature and humidity.
- Cloud vs. Local Processing:
    a. *Tradeoff:* Offloading data processing to a cloud server may provide scalability but introduces latency and dependence on internet connectivity.
    b. Decide whether to perform data processing locally on the computer or leverage cloud services. Local processing might be preferred for real-time responsiveness, while cloud processing offers scalability and the ability to analyze historical data.
- User Customization vs. Simplicity:
    a. *Tradeoff:* Allowing users to customize the app based on personal preferences introduces complexity, potentially affecting the app's simplicity and ease of use.
    b. Consider providing customization options such as temperature units (Celsius or Fahrenheit) and humidity thresholds for alerts. However, carefully design the user interface to maintain simplicity and avoid overwhelming users with choices.

Tech Stack:

- TTGO T-Watch: Program the T-Watch using Arduino IDE
- Computer: Use same framework in class, making use of AWS EC2 and similar tools
- Weather and Humidity Sensor: Reads temperature and humidity and serves it back
- Connect to phone using wifi, and casting it

Expected Challenges:

- Data Synchronization: Implement robust data synchronization mechanisms to ensure consistency across devices, handling potential issues like packet loss.
- Security: Employ encryption methods for data transmission to protect user information.
- Power Consumption: Experiment with different transmission frequencies and power-saving modes on the T-Watch to optimize battery life.
- Simply figuring out how to cast the information to my phone will be difficult, I’ve never used technology like that before, need to come up with a good way to do that.
- Additionally, need to understand the tech stack fully and how this can be implemented in future mobile applications.

Task Sharing Timeline

Afternoon:

- Develop a minimal T-Watch set-up to simulate data transmission from sensor
- Begin setting up on the computer to receive simulated data.
- Finalize the basic communication between the T-Watch and the computer (using simulated data).
- Start a simple mobile app development (perhaps a basic UI framework).

Nigh:

- Complete a rudimentary mobile app UI.
- Integrate the mobile app with the computer for simulated data reception.
- Conduct a quick test to ensure basic functionality