Offline Wireless Ai Project

Thursday, 5 June 2025 10:40 pm

Project Title:
Design and Implementation of an Offline Wireless Communication System with AI-Based Signal Quality Optimization

Project Summary:
This project involves building a wireless communication network using ESP8266 microcontrollers that allows users to send and receive messages without the need for internet. The system uses real-time signal data (like RSSI and latency) to determine the best path for message delivery, with AI assisting in the routing decision. Messages hop between devices (nodes), and the routing is optimized for performance based on signal conditions.

Key Components:
- ESP8266 modules as wireless communication nodes
- AI model
- Communication via ESP-NOW or Wi-Fi Direct (no internet)
- RSSI and latency measurement
- Optional encryption for message privacy

System Flow:
1. Node A (sender) broadcasts a message intended for Node D (receiver)
2. Nearby nodes (Node B, Node C) receive the broadcast and measure RSSI and latency
3. These nodes send signal data to the central AI node (laptop)
4. The AI compares signal quality and selects the best node to forward the message
5. Node A sends the full message to the selected relay node (e.g., Node B)
6. Node B forwards it to Node D

Example Message Format:
```
{
 "from": "Node A",
 "to": "Node D",
 "data": "Encrypted message here",
 "msg_id": 001
}
```

Node Roles:
- Each ESP8266 is a node (held by a person)
- Roles: Sender, Relay, Receiver
- MAC addresses are used to identify nodes

AI Role:
- Receives input: RSSI, latency, success rate
- Outputs: Best node to relay the message
- Trained in Python

Tai's Focus:
- Wireless communication
- Signal metrics: RSSI, latency, packet loss
- Setting up and testing ESP8266 network

Delano's Focus:

- AI model selection and training
- Prediction logic for routing
- Integration with central laptop

Build Phases:

1. Flash sender and receiver test (ESP-NOW)
2. Log RSSI and latency on receive
3. Add 3rd/4th nodes for multi-hop
4. Send signal data to laptop
5. Train AI with collected data
6. Laptop tells ESP which path to use
7. Relay forwards message to destination

Languages & Tools:

- Arduino C++ for ESP8266
- Python for AI (scikit-learn)
- Serial/Wi-Fi for ESP-laptop connection