

## **ECE435: Embedded Systems – Final Project**

**Due: Friday, 8 May 2026 (Last day of Finals)**

### **Overview:**

- Design a project that in some way involves a network.

### **Guidelines:**

- You may work either alone or in groups of two or three. If you work in a group your end project will have higher expectations.
- You may use any hardware you like, as long as it is network related.
- You may use any programming language you like.

### **Part 1: Topic Selection** (due Friday 27 March 2026) (5pts)

Each group should send a brief e-mail describing your project topic and listing group members.

### **Part 2: Progress Report** (due Friday 17 April 2026) (10pts)

A brief status update detailing progress your group has made. This is primarily to make sure your project is on track to be finished in time; if things are not going well the topic can be adjusted.

Send this report by e-mail. Only one submission is needed per group.

1. State in one sentence a summary of your project.
2. Describe the hardware/software that you will be using
3. Have you done any preliminary testing yet? Are you on track for being finished on time?
4. What day would you like to present? Monday (27 April) Wednesday (29 April) or Friday (1 May)? There will be some bonus points for presenting early.
5. Submit the status update by e-mail.

### **Part 3: In-Class Presentation** 27 April & 29 April & 1 May 2026 (40pts)

- You will have roughly 10 minutes to present (there might be less depending on how many presentations are scheduled that day). Be sure to budget time for questions and demo (8 minutes of presenting, 2 minutes of demo/questions?) Points may be taken off for going over.
- You may present slides using the projector if you want, but that's not strictly necessary.
- Your presentation should have at least the following information. Feel free to include more.
  - Brief overview of your project.
  - A summary of the hardware/software being used
  - Related work (briefly list any similar projects you are aware of)
  - What network layers are involved
  - Any security issues with the project
  - Any ethical concerns with the project
  - Challenges: list any challenges you had getting things working.
  - Future work: things you might add if you had more time.
  - Leave time to do a brief demo if possible

### **Part 4: Project Writeup**, Due 8 May 2026 (45pts)

This will be a short paper (at least 6 pages, but you can include pictures, diagrams, etc.) that must contain all of the following:

1. Introduction: The purpose of your project and a high level overview.
2. Related Work
  - (a) Has anyone done a project like this before?  
If so describe/cite it. If not, what's the most similar thing you can find?
  - (b) How does your project compare to the existing similar projects?
3. Experimental Setup:
  - (a) Describe any hardware you used
  - (b) Describe any software you used  
(for hardware and software, try to include enough information that someone could replicate your work)
  - (c) What network layers are involved?
  - (d) Are there any security issues with your project?
  - (e) Are there any ethical concerns with your project?
4. Results:

- (a) What results did you find?  
Tables and Graphs can be nice.

## 5. Conclusion

- (a) If you worked in a group: List who worked on what part.
- (b) Challenges: List any challenges you had, and if things didn't work, explain why.
- (c) Future Work: List any improvements you might make if you had more time and resources to work on the project.

## 6. Appendix

- (a) Any source code (this can be submitted as a separate file, does not have to be included in the report).

E-mail your final report to me. pdf or word document is fine, the code should be attached too.

## Project Ideas:

- Physical Layer
  - GNU radio, software-defined radio
    - Use SDR to capture network packets
    - Use SDR to monitor airplane flights
    - Use SDR to download images from satellites
  - Compare fiber vs wired vs wireless
- Link Layer
  - Explore limits of Bluetooth
  - Connect two Raspberry Pis over bluetooth?
  - Use a Raspberry Pi to interface with bluetooth devices (wii controller? headset?)
  - Xbee/Zigbee
  - HAM radio networking
  - Use bluetooth near 802.11 and see if packet rates go down due to interference?
  - How does 802.11 bandwidth/latency go down with distance?
  - Triangulate position based on nearby wireless routers and signal strength
- Network / Internet Layer
  - Evaluate different TCP congestion algorithms
  - Set up routing in a Network Simulator
  - Set up a IPv6 network between some raspberry pis
    - Compare to IPv4 setup
  - Write simple IP stack for small embedded board (Apple II, Arduino, etc)

- Packet intercepting. Project where wireless router acts as proxy, flips all images upside down
- Benchmark. What's the fastest bandwidth you can get out of your machine? Out of a Raspberry Pi?
- fastest bandwidth you can get of a gigabit Ethernet on windows/Linux/osx?
- Multicasting. Multicast something?
- Application Layer
  - Extend the homework webserver to be TLS, http2 or http3 compliant
  - Write a web browser (this is very hard!)
  - Set up a webserver on a Raspberry Pi and do something interesting with it
  - Write a networked video game, chat room, etc
  - Set up network attached storage
  - Benchmark various web-servers.
  - Write some code using web-sockets.
- Security
  - Explore (on a private network) various DoS methods and how to block them
  - Set up a firewall with advanced features
  - Set up pi-hole ad blocking software

### **Project Ideas from Previous Years:**

- http2 webserver
- Canbus networking
- Custom QUIC implementation
- Video game network tuning
- Bluetooth connected Raspberry Pis
- Pi-hole ad blocking
- Webserver benchmarking
- Pi webcam
- SDR monitoring airline flights
- Online checkers game
- UDP audio streaming
- Pi webserver guestbook
- ARP spoofing

- Bluetooth light sensor
- Weather station with SQL/Web interface
- Network-connected robot
- Networked video game
- Networked parking sensor
- DNS server setup
- Ping pong score server
- VPN setup
- Text-based web browser
- Custom Ethernet driver in FPGA
- Encrypted network tunnel
- Raspberry Pi network chat
- Unity to python client/server interface
- Raspberry Pi DDoS mitigation
- Networked Battleship game
- Raspberry Pi network attached storage