# Open-minded Project:

**1. Implement a Simple Web Server:** Design and implement a simple web server based on the TCP/IP protocol. You can handle HTTP requests, transmit static resources, generate dynamic content, and implement basic session management and security controls.

**2.Develop a Chatroom Application:** Implement a multi-user chatroom application that allows users to chat in real-time on the same network. You can use socket programming to handle communication between clients and the server, and implement features such as message delivery, user management, and chat log storage.

**3.Network Protocol Simulator:** Develop a network protocol simulator for simulating and testing the behavior of different network protocols. You can implement generation and parsing of protocol packets, simulate network topologies and link conditions, and analyze protocol performance and reliability.

**4.Implement a Simple FTP Client/Server:** Design a file transfer client and server based on the FTP protocol. You can handle file upload and download requests, implement file storage and management, and add security mechanisms such as authentication and encryption.

**5.Implement a Simple VPN Client/Server:** Design and implement a virtual private network (VPN) client and server based on a VPN protocol. You can use encryption and authentication to protect network communication and support remote access and secure data transmission.

**6.Network Performance Testing Tool:** Develop a network performance testing tool for measuring network latency, bandwidth, and throughput. You can implement ping tests, bandwidth measurement, traffic generation, and generate performance reports and charts.

**7.Implement a Simple DNS Server:** Design and implement a simple DNS server based on the DNS protocol. You can handle domain name resolution requests, implement domain-to-IP address mapping, and support high-performance caching and load balancing.

**8.Implement a simple online collaborative editor:** Design and implement an online collaborative editor that allows multiple users to perform real-time collaborative editing on the same document. You can use socket programming to handle communication between users and achieve real-time text synchronization and coordinated editing operations.

**9. Implement a simple online whiteboard tool:** Design and implement an online whiteboard tool that allows multiple users to perform real-time drawing and marking on the same canvas. You can use socket programming to handle communication between users and achieve synchronization and display of drawing operations.

**10. Implement a simple multiplayer quiz game:** Design and implement a multiplayer quiz game where one player serves as the host and presents questions to other players. Other players submit their answers through the network. You can use socket programming to handle the transmission of questions and answers and implement score calculation and game flow control.

# Requirement:

1. **Group:** 1-5 students one group;
2. Choose any one of the projects to implement;
3. Upload following material to Moodle:
   (1) Source code;
   (2) The demo of project (video format);
   (3) Project report: The project must be written in English, otherwise 0 points will be given. The content can include but not limited to project requirement analysis、detail implementation of project, results of project.