

# Technical Specification for Network Application Development\*\*

---

## Technical Specification for Network Application Development

### 1. Development of a Client-Server Application in Python for File Transfer

- **Objective:** To create a Python application that enables a client to send files to a server using the TCP protocol.
- **Functional Requirements:**
  - Implementation of a server component capable of accepting connections and files from clients.
  - Implementation of a client component to send files to the server.
  - Ensuring the integrity and confidentiality of the transferred data.
- **Non-Functional Requirements:**
  - User-friendliness.
  - High reliability of data transmission.
- **Deadline:** December 24th.

### 2. Network Topology Assembly in Mininet and Application Testing

- **Objective:** To assemble a network topology with various latency (RTT), bandwidth (BE), and packet loss rate (PLR) parameters, and to run the developed application in this environment.
- **Functional Requirements:**
  - Development of topology using Mininet tools.
  - Configuration of network parameters according to specified characteristics.
  - Conducting file transfer testing in the developed network environment.
- **Deadline:** January 15th.

### 3. Adaptation of the Transmission System to the Ant Colony Algorithm

- **Objective:** To integrate the Ant Colony Algorithm into the system for optimization of routing and calculation of the shortest path.
- **Functional Requirements:**
  - Implementation of the Ant Colony Algorithm for dynamic route selection.
  - Testing and assessing the efficiency of the new routing system.
- **Deadline:** February 1st.

### 4. Implementation of Application Layer Multicast in Mininet Environment

- **Objective:** Based on the adapted data transmission system, implement an application layer multicast mechanism in the established network topology.
- **Functional Requirements:**
  - Development of a multicast mechanism for effective data distribution to multiple recipients.
  - Integration of the mechanism into the existing system and Mininet topology.
  - Testing to evaluate the performance and reliability of the mechanism.
- **Additional Requirements:**
  - Configurational flexibility.
  - Compatibility with various client and server systems.
- **Deadline:** February 15st.

All the above tasks must be carried out on a Linux operating system, ensuring compatibility and utilization of the Linux environment's features and tools. Each stage should have a detailed implementation plan that includes hardware and software requirements, as well as acceptance criteria and testing procedures.