PROJECT ABSTRACT

LAN File Transfer Desktop App with Drag-and-Drop Sharing

A desktop application for fast, encrypted file sharing over a local network: drag-and-drop send queue, automatic LAN device discovery, 1 MiB SHA-256-hashed chunks over TLS 1.3, auto-resume after interruption, and live throughput graphs. Built to order with source, protocol documentation, demo and complete viva kit.

01. Title

LAN File Transfer Desktop App with Drag-and-Drop Sharing

02. Introduction / Background

Moving large files between lab computers still means USB drives, email attachments, or uploading to the cloud and downloading again — slow, and every hop leaves a copy somewhere else. On a local network, machines can talk to each other directly at full LAN speed, but students rarely get to build that path themselves: socket programming, peer discovery, chunked transfers with integrity checks, and resume logic are usually just theory. This project turns them into a working desktop app. Files are split into 1 MiB chunks, each hashed with SHA-256, streamed over a TLS-encrypted TCP connection to a peer discovered on the LAN, and reassembled at the other end. If the connection drops, the transfer resumes from the last confirmed chunk instead of starting over.

03. Problem Statement

Existing file-sharing routes — USB drives, email, cloud uploads — are slow and scatter copies across third-party servers, while the core networking concepts behind a proper local transfer (TCP framing, mDNS-style discovery, chunked integrity, resume-after-interrupt) stay theoretical for most students. This project addresses both gaps: a desktop application that transfers files directly between LAN peers with integrity verification, encryption and resume, demonstrating applied networking through a usable, demonstrable product.

04. Objectives

- Discover peers automatically on the LAN with mDNS-style broadcast discovery — no manual IP entry.
- Transfer files in 1 MiB chunks, each verified with a SHA-256 hash before being written to disk.
- Resume interrupted transfers from the last acknowledged chunk instead of restarting.
- Encrypt all transfer traffic with TLS 1.3 so files cannot be sniffed on shared networks.
- Provide a drag-and-drop send queue with per-file progress, live speed and pause/cancel controls.
- Render an aggregate throughput chart plus a history log of every transfer.
- Ship protocol documentation, a working demo and the full viva kit (report, PPT, Q&A;).

05. Proposed Methodology

The app is built in three layers. (1) Discovery: on launch, each instance broadcasts its presence on the LAN and listens for peers, building the device list without manual configuration. (2) Transfer protocol: each file is split into 1 MiB chunks; a manifest with file name, size and chunk hashes is sent first over a TLS 1.3 TCP connection on the configured port, then chunks stream to the peer, which acknowledges each one after verifying its SHA-256 hash. (3) Presentation: a send queue shows per-file progress and chunk counters, a live throughput chart aggregates speed, receive requests need an explicit accept (unless the sender is trusted with auto-accept), and a history log records every completed transfer with timestamps.

06. System Architecture / Working

The user drops files into the send queue and selects a destination device from the discovered sidebar. The file is chunked, and the manifest travels first over a TLS 1.3 TCP connection. Chunks stream to the peer, which acknowledges each one and verifies its hash before writing to disk; corrupted chunks are re-requested. If the connection breaks, both sides remember the last acknowledged chunk and resume from there on reconnect. The receiver gets an accept/decline prompt, completed files land in the configured save folder, and every transfer — sent or received — is recorded in the history log with peer, size, direction and timestamp.

07. Hardware / Software Requirements

- Desktop OS: Windows, macOS or Linux (expected platforms)
- Python 3 with PyQt or Tkinter for the desktop UI (desktop build)
- TCP sockets with TLS 1.3 support
- A local network permitting LAN broadcast/multicast for discovery
- Modern web browser for the single-file HTML/CSS/JS demo build
- Git

08. Expected Outcomes

- A working desktop app that transfers files between LAN peers with per-chunk integrity checks.
- Automatic device discovery with no manual IP configuration.
- Reliable auto-resume from the last acknowledged chunk after a dropped connection.
- Live throughput charts and a complete transfer history log.
- Protocol design documentation and a full viva kit: report, PPT and Q&A; on sockets, TCP framing, TLS, hashing and discovery.

09. Applications

- Fast file exchange between lab, classroom and hostel computers without the internet.
- Teaching socket programming, peer discovery, chunked transfers and checksums.
- Demonstrating TLS-encrypted peer-to-peer communication in coursework.

10. Future Scope

- NAT traversal / internet relay for transfers beyond the LAN.
- Multi-peer swarm-style transfers with parallel chunk sourcing.
- Mobile companion app (Android) for phone-to-desktop sharing.

- Bandwidth scheduling and time-of-day transfer rules.