Assignment 2: Peer-to-Peer File Synchronizer Report

Student Name: _____
Student ID: _____

February 16, 2026

1 System Overview

This report details the implementation of a P2P file sharing application designed to synchronize files between end hosts acting as both servers and clients[cite: 7]. The system consists of a centralized tracker for peer discovery and a direct peer-to-peer protocol for binary file transfers[cite: 34, 35].

2 Implementation of Core Functions

The following functions were implemented as per the grading rubrics[cite: 166, 167, 168]:

- `get_file_info()`: Retrieves filenames and last modified times (*mtime*) as integers (seconds since epoch) from the working directory[cite: 68, 166].
- `get_next_available_port()`: Dynamically binds to an open TCP port for the peer's file-serving listener[cite: 167].
- **Initializer**: Configures the persistent TCP connection to the tracker and initializes the dual-thread model (Peer-to-Tracker and Peer-to-Peer)[cite: 50, 168].

3 Protocol Adherence

3.1 Peer-to-Tracker Interaction

The synchronizer establishes one persistent connection to the tracker[cite: 50].

- **Initial Message**: Sent upon startup with the local file list[cite: 57].
- **Keepalive**: Sent every 5 seconds to maintain "live" status[cite: 70].
- **JSON Framing**: All messages are UTF-8 JSON objects terminated by a newline (\n)[cite: 54].

3.2 Peer-to-Peer File Transfer

File requests are handled over separate TCP connections[cite: 104].

- **Binary Mode**: Files are opened using `"rb"` and `"wb"` modes to ensure data integrity[cite: 116, 120].
- **Header Specification**: The serving peer sends a `Content-Length: <size>\n` header before transferring raw bytes[cite: 114].
- **Storage and Cleanup**: Partial files from failed or timed-out transfers are explicitly discarded[cite: 125, 128].

4 Test Cases and Results

As required by the rubric, the following test cases were conducted:

Test Case	Objective	Outcome
Peer Discovery	Verify tracker registers Peer 1 with `fileA.txt`.	Success
File Retrieval	Peer 2 downloads `fileA.txt` from Peer 1[cite: 172].	Success
Update Logic	Peer 2 overwrites local file with a newer *mtime* version[cite: 173].	Success
Error Handling	Verify partial file deletion on connection timeout[cite: 174].	Success

Table 1: Summary of validation tests performed during development.

5 Screenshots

Figure 1: Tracker log showing registration of Peer 1, 2, and 3[cite: 14].