**Server Implementation**

pip install qrcode[pil]

import socket

import os

# Configuration

HOST = '0.0.0.0' # Listen on all interfaces

PORT = 65432 # Port to listen on

# Ensure a directory for files exists

if not os.path.exists('shared\_files'):

os.makedirs('shared\_files')

def handle\_client(conn):

while True:

# Receive command from client

command = conn.recv(1024).decode()

if not command:

break

# Handle file upload

if command.startswith("UPLOAD"):

filename = command.split()[1]

file\_size = int(command.split()[2])

with open(f'shared\_files/{filename}', 'wb') as f:

bytes\_received = 0

while bytes\_received < file\_size:

data = conn.recv(1024)

if not data:

break

f.write(data)

bytes\_received += len(data)

conn.send(b"UPLOAD\_COMPLETE")

# Handle file download

elif command.startswith("DOWNLOAD"):

filename = command.split()[1]

if os.path.exists(f'shared\_files/{filename}'):

conn.send(b"FILE\_FOUND")

with open(f'shared\_files/' + filename, 'rb') as f:

data = f.read(1024)

while data:

conn.send(data)

data = f.read(1024)

conn.send(b"DOWNLOAD\_COMPLETE")

else:

conn.send(b"FILE\_NOT\_FOUND")

else:

conn.send(b"INVALID\_COMMAND")

conn.close()

def start\_server():

with socket.socket(socket.AF\_INET, socket.SOCK\_STREAM) as s:

s.bind((HOST, PORT))

s.listen()

print(f"Server listening on {HOST}:{PORT}")

while True:

conn, addr = s.accept()

print(f"Connection from {addr}")

handle\_client(conn)

if \_\_name\_\_ == "\_\_main\_\_":

start\_server()

**Client Implementation**

import socket

import os

import qrcode

HOST = '127.0.0.1' # Server's hostname or IP address

PORT = 65432 # The same port as the server

def upload\_file(filename):

if os.path.exists(filename):

file\_size = os.path.getsize(filename)

with socket.socket(socket.AF\_INET, socket.SOCK\_STREAM) as s:

s.connect((HOST, PORT))

s.send(f"UPLOAD {os.path.basename(filename)} {file\_size}".encode())

with open(filename, 'rb') as f:

bytes\_sent = 0

while bytes\_sent < file\_size:

data = f.read(1024)

s.send(data)

bytes\_sent += len(data)

print(s.recv(1024).decode()) # Receive confirmation

else:

print("File does not exist.")

def download\_file(filename):

with socket.socket(socket.AF\_INET, socket.SOCK\_STREAM) as s:

s.connect((HOST, PORT))

s.send(f"DOWNLOAD {filename}".encode())

response = s.recv(1024).decode()

if response == "FILE\_FOUND":

with open(f'downloaded\_{filename}', 'wb') as f:

while True:

data = s.recv(1024)

if data == b"DOWNLOAD\_COMPLETE":

break

f.write(data)

print(f"Downloaded {filename}")

else:

print("File not found on server.")

def generate\_qr\_code(url):

qr = qrcode.make(url)

qr.save('file\_link\_qr.png')

print("QR Code generated and saved as 'file\_link\_qr.png'")

if \_\_name\_\_ == "\_\_main\_\_":

choice = input("Enter 'upload' to upload a file or 'download' to download a file: ")

if choice == 'upload':

filename = input("Enter the file path to upload: ")

upload\_file(filename)

# Generate a QR code for the file link (example)

generate\_qr\_code(f"http://{HOST}:{PORT}/shared\_files/{os.path.basename(filename)}")

elif choice == 'download':

filename = input("Enter the file name to download: ")

download\_file(filename)

**Running the Application**

1. Start the server in one terminal:

python server.py

2. Start the client in another terminal:

python client.py