

Week 2 Checklist: Handle Multiple Clients

Goal

Extend the server to support multiple clients. Use threading or I/O multiplexing to handle multiple simultaneous connections. Update the README and test the new functionality.

1. Threading or Multiplexing

- Choose a strategy: `pthread` (threads) or `select()` (I/O multiplexing)
- If using threads:
 - Create a new thread for each client using `pthread_create()`
 - Detach or join threads appropriately
- If using `select()`:
 - Add listening socket and client sockets to fd set
 - Use `FD_SET`, `FD_ISSET`, and `select()` loop

2. Server Logic Changes

- Accept multiple client connections in a loop
- Ensure each client gets independent handling
- Add printing/logging for new connections
- Ensure clean handling of client disconnects

3. Testing

- Run multiple `telnet` or `nc` sessions simultaneously
- Confirm each client can send and receive messages independently
- Ensure server does not crash or hang with more than one client
- Validate graceful handling of one client disconnecting while others remain

4. Documentation

- Update `README.md`:
 - Note new multi-client support
 - Explain which strategy was used (`pthread` or `select`)
 - Update testing instructions
- Commit code and README updates

5. Done

- Push updates to GitHub
- Mark Week 2 complete in project plan
- Note any bugs or limitations to fix in Week 3