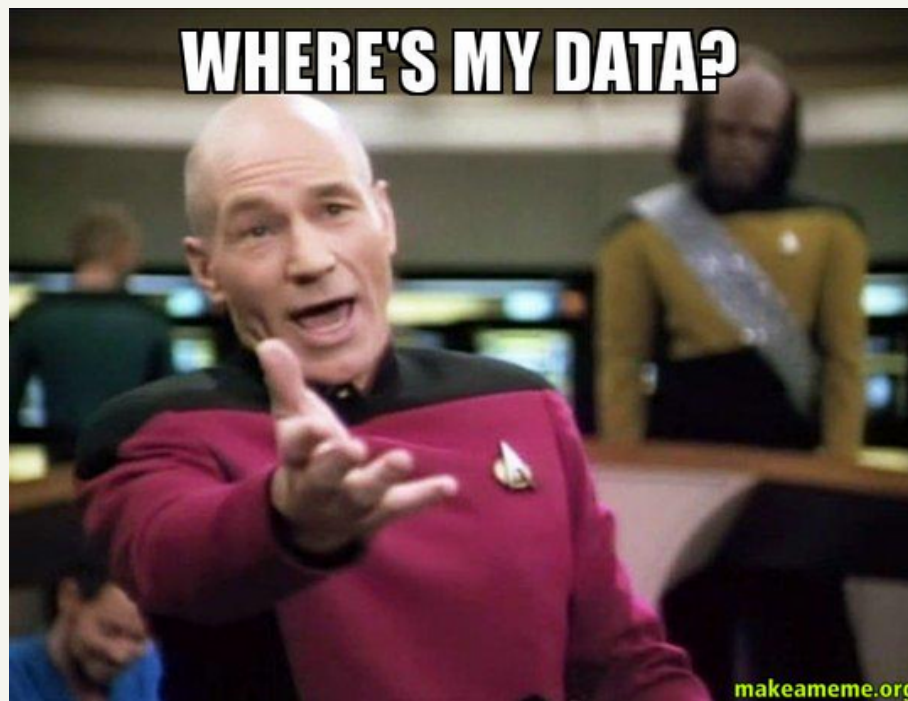# Operating System
Repository Synchronization

— 01

**Final Presentation
and Working Demo**

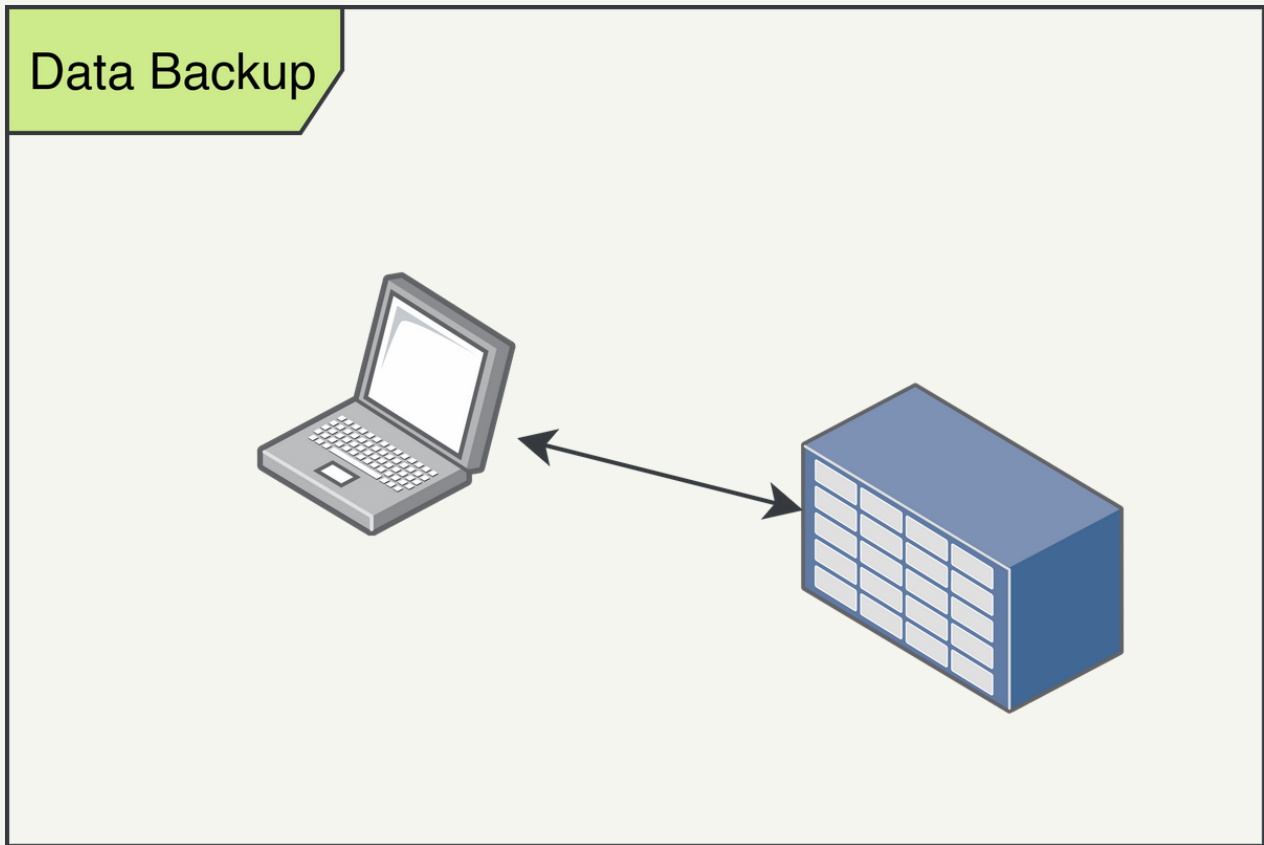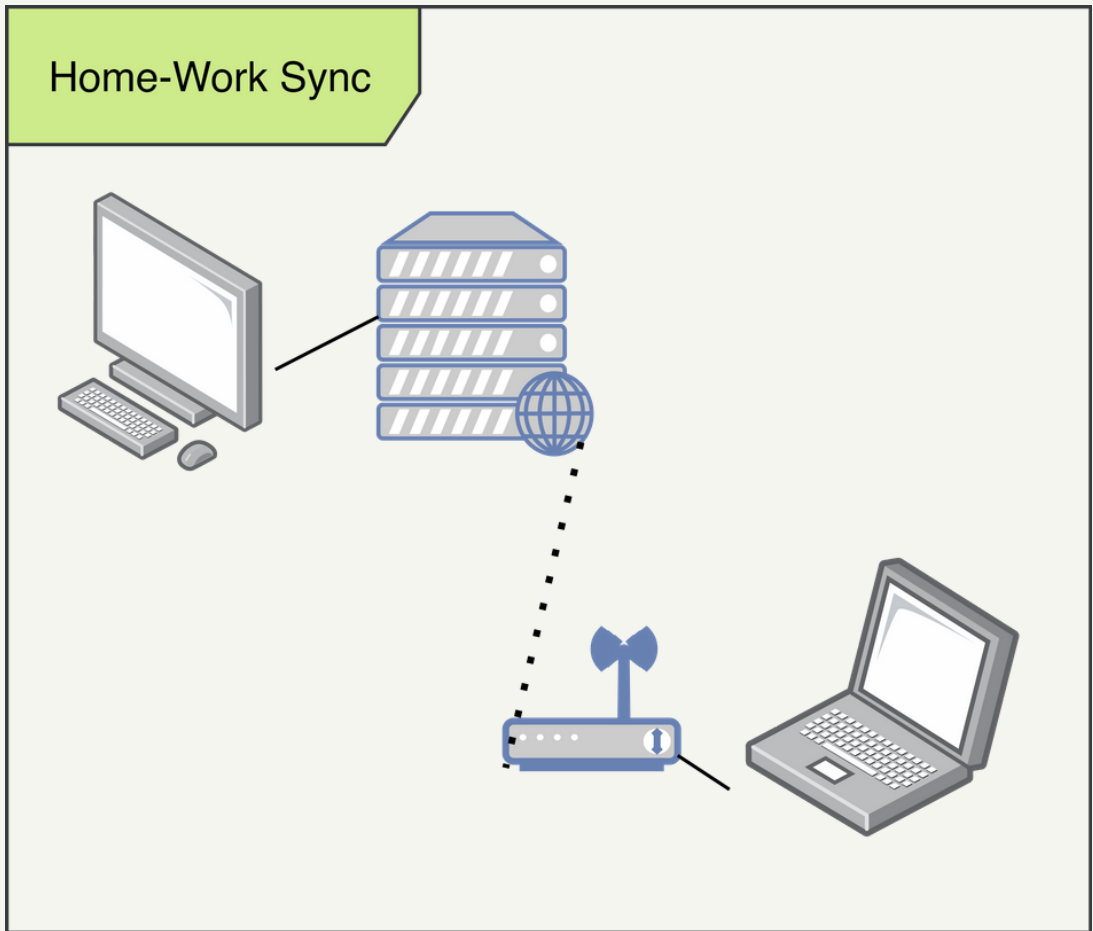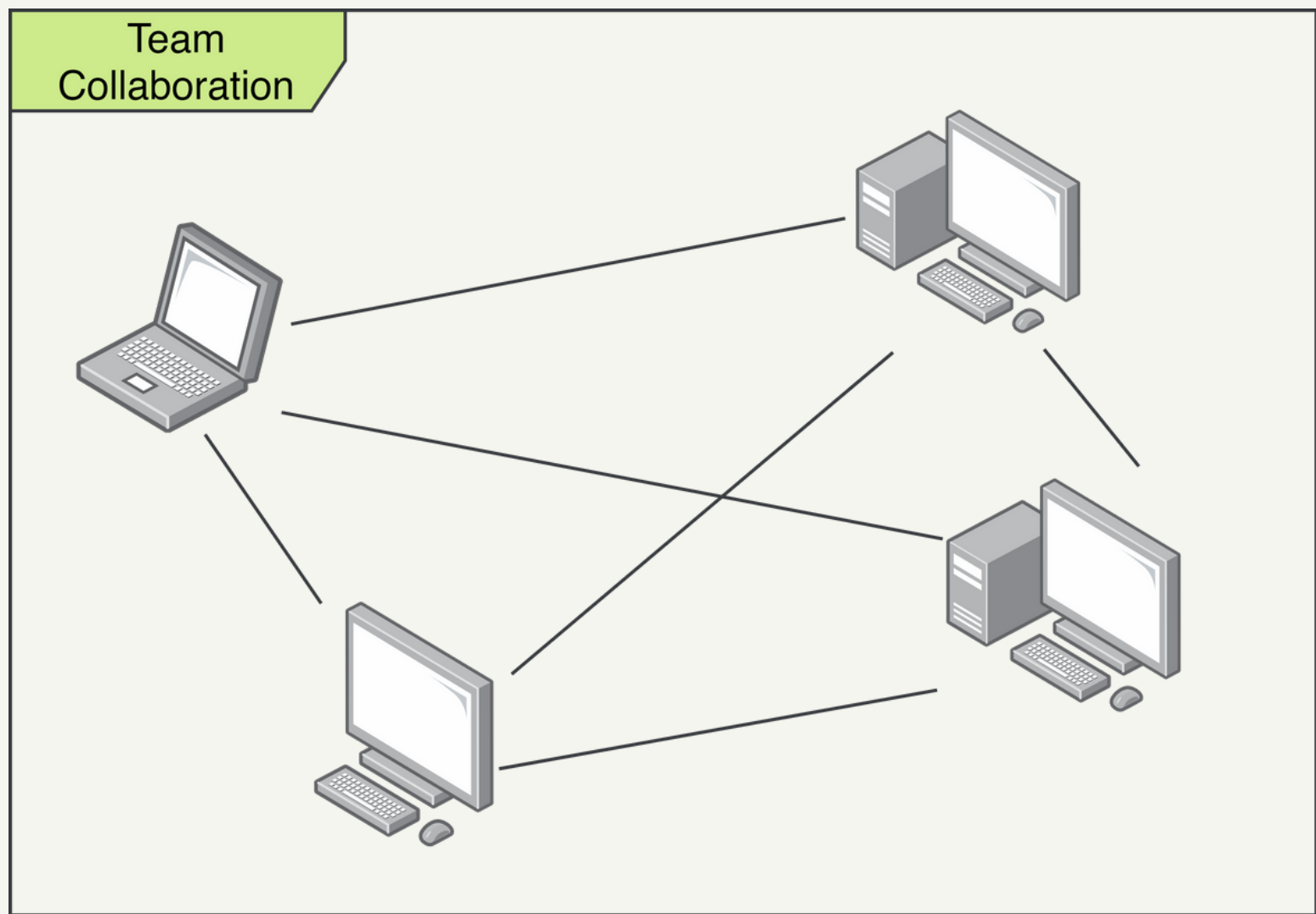# Motivation

- Want our data to move with us
- Work without disturbing others workflow
- Sychronize the work of every team member

WHERE'S MY DATA?

makeameme.org

Backup & Sync

Need of Project

Team Collaboration

Home-Work Sync

Data Backup

Related Work

Feature Set | Objectives

- Properly be able to synchronize folders

- Handle any number of users connected on LAN

- Peer2Peer mode: Decentralized system

- Low Latency

- Robust & Expandable Product

Exhibit the goals of Multi-Threading
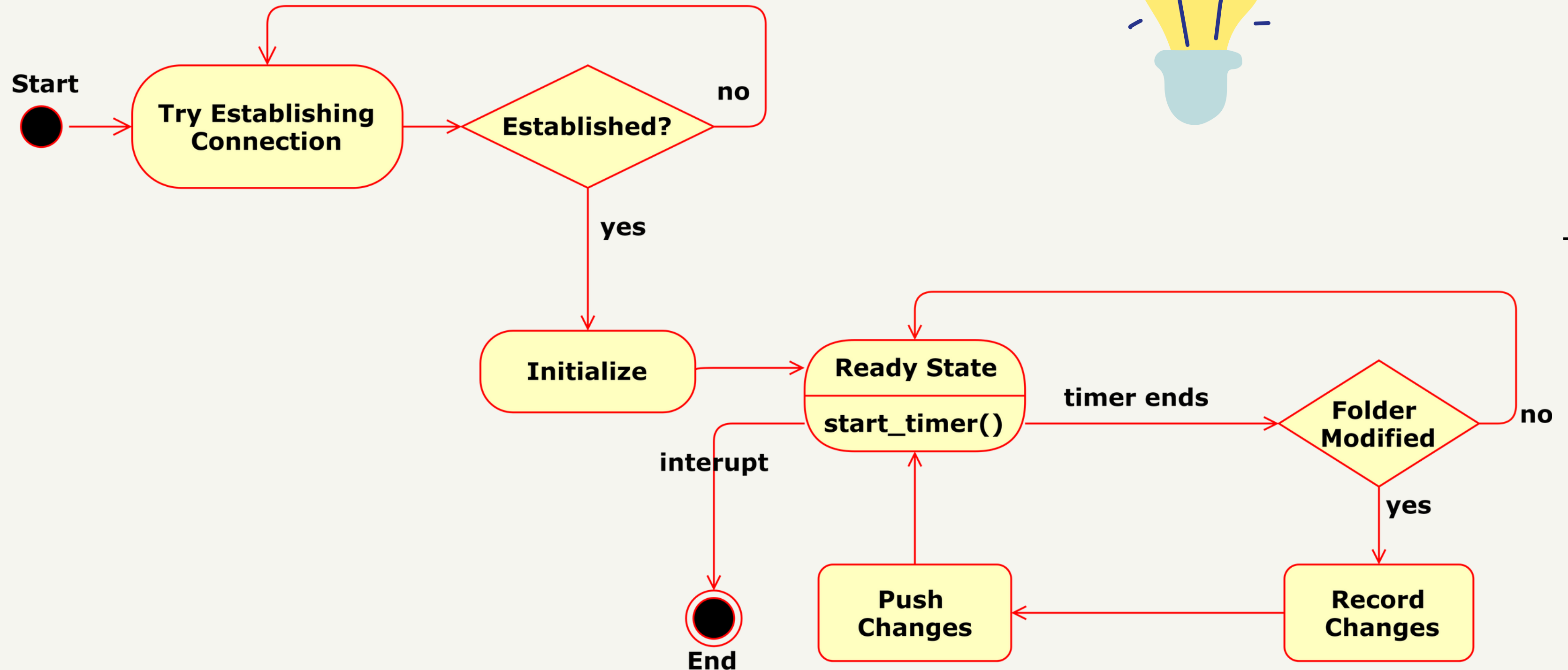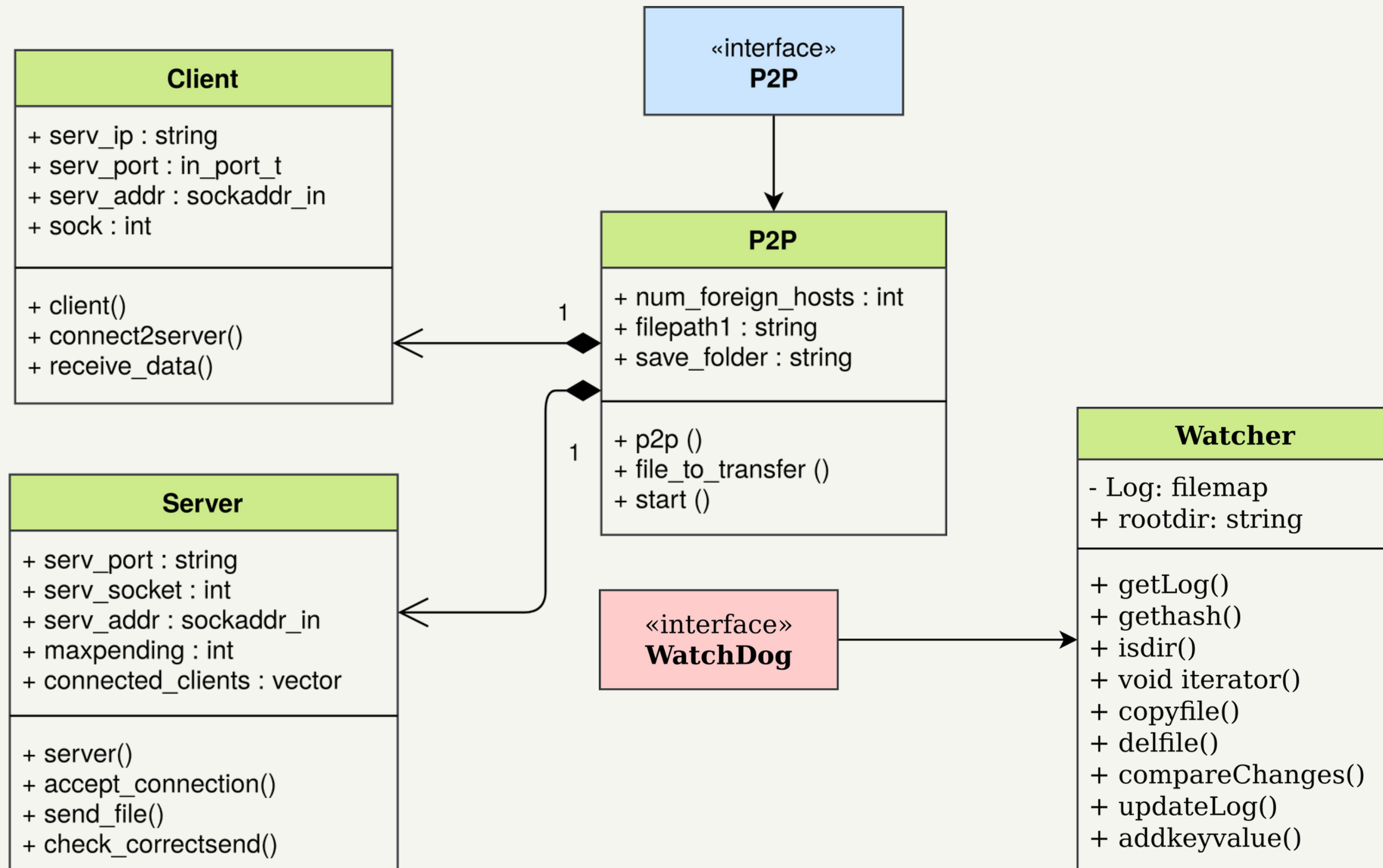
Logic Flow | Intuition

Start

Try Establishing Connection

Established?

no

yes

Initialize

Ready State

start_timer()

timer ends

Folder Modified

no

yes

interupt

End

Push Changes

Record Changes

Code Structure

Client

+ serv_ip : string
+ serv_port : in_port_t
+ serv_addr : sockaddr_in
+ sock : int

+ client()
+ connect2server()
+ receive_data()

«interface»
P2P

P2P

+ num_foreign_hosts : int
+ filepath1 : string
+ save_folder : string

+ p2p ()
+ file_to_transfer ()
+ start ()

1

1

Server

+ serv_port : string
+ serv_socket : int
+ serv_addr : sockaddr_in
+ maxpending : int
+ connected_clients : vector

+ server()
+ accept_connection()
+ send_file()
+ check_correctsend()

«interface»
WatchDog

Watcher

- Log: filemap
+ rootdir: string

+ getLog()
+ gethash()
+ isdir()
+ void iterator()
+ copyfile()
+ delfile()
+ compareChanges()
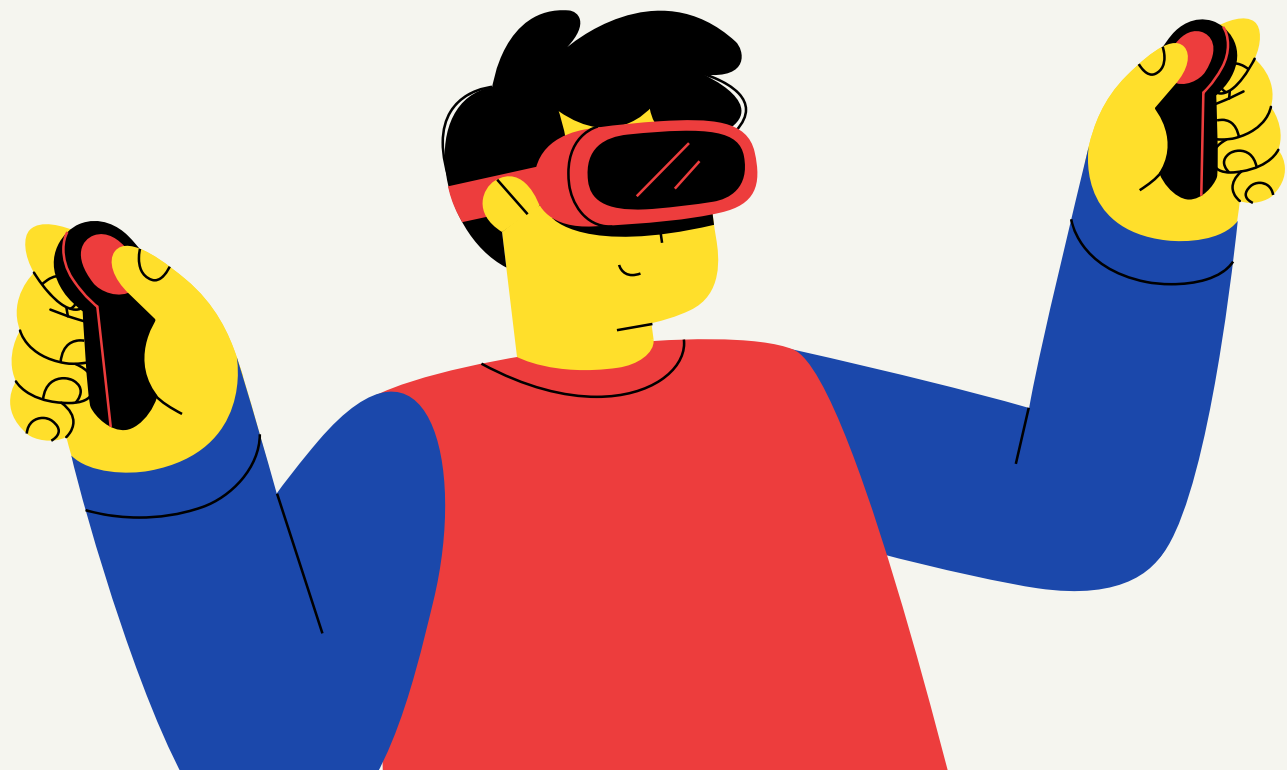+ updateLog()
+ addkeyvalue()

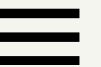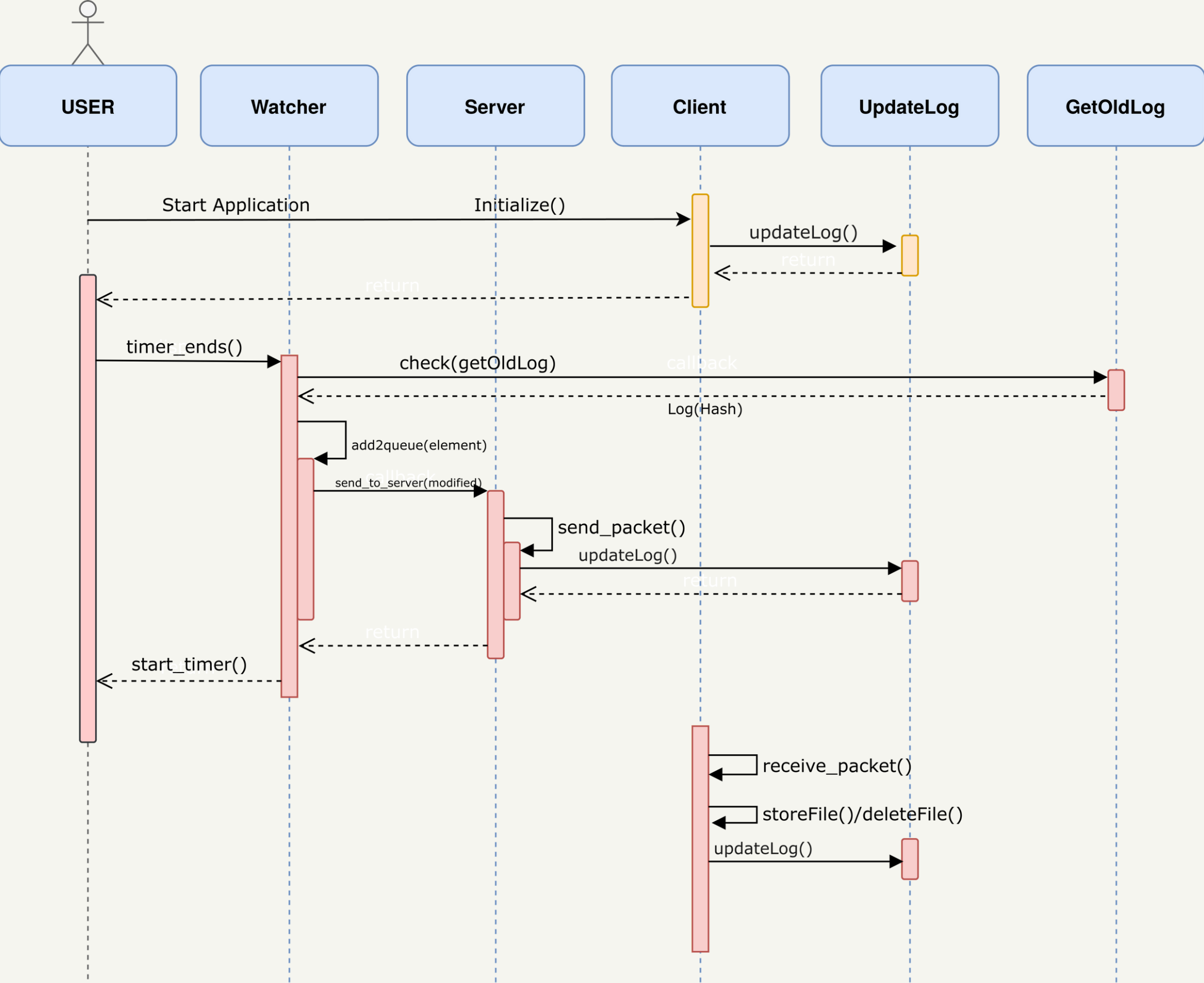# Ticket Lock

Fairness

Mutually Exclusion

No Starvation

Working Demo

Code Flow

LIMITATIONS

- Doesn't allow dynamic peer connections and disconnections.
- Simultaneous modification in same file by multiple peers would cause a race condition.

FUTURE WORK

- Support for Cross-Platform Synchronization.
- Incremental updating of files.
- Version Control

Thankyou