Assignment #3

Multi-Client Chat Program

Steffen L. Norgren
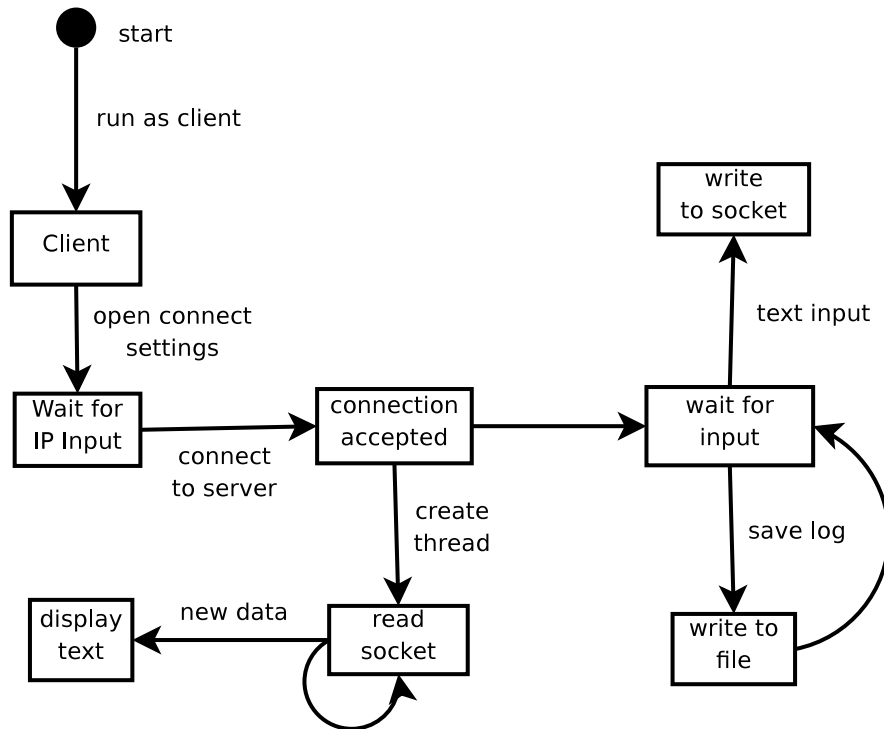A00683006

COMP 4981 • BCIT • March 18, 2009w

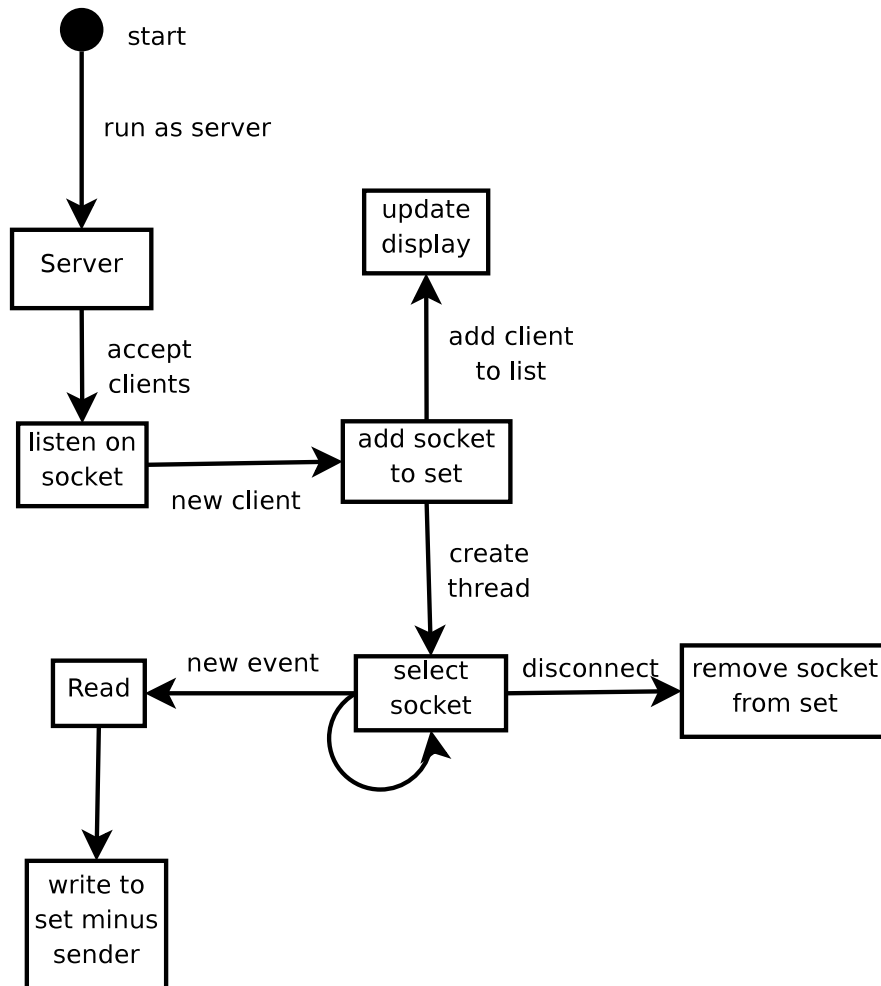# TABLE OF CONTENTS

APPLICATION DESIGN & DIAGRAMS

Client

start

run as client

Client

open connect
settings

Wait for
IP Input

connect
to server

connection
accepted

create
thread

wait for
input

write
to socket

text input

save log

write to
file

read
socket

new data

display
text

start

run as server

Server

accept
clients

update
display

add client
to list

listen on
socket

new client

add socket
to set

create
thread

new event

Read

select
socket

disconnect

remove socket
from set

write to
set minus
sender

PSEUDO-CODE

Client

```
client-connect start

    check if IP/hostname is valid

    attempt to open a connection on the socket

    if connect failed then

        print error


    create read-socket thread

    create user-input thread

client-connect end

read-socket begin

    while true

        read from socket

        if data exists

            write to display

read-socket end

user-input begin

    while true

        if user input then

            if write log to disk then

                write user chat log to local directory

            if request to send data then

                write data to socket

user-input end
```

Server

```
begin server-start

      create listening socket

      if new connection

            add socket to master list

            create read-socket-list thread

end server-start

begin read-socket-list

      while true

            for each socket in master list

            select(socket, event)

            if event is FD_ISSET

                  add new client connection to master list

                  write client hostname to GUI

            if event is other

                  read from socket in set

                  write data to all in set minus current socket

end read-socket-list
```

TESTING

Testing for this application was relatively straight forward, as I decided to create it as a GUI rather than a command-line program. If it were a command-line driven application, I could have redirected input to the client to test throughput and such. Unfortunately I don't know how to do that yet within a GUI.

For the most part, testing consisted of simply copying and pasting strings into the client's input as well as testing multiple client connections and disconnects. Though testing I discovered a few bugs that I was unable to correct.

Bug #1

When one of the GTK+ windows does not have focus, it takes a long time for it to update its display when new text arrives. One can get around this bug by moving the cursor over the window, but this is obviously not ideal. Being that I am new to programming with GTK+, I was unable to find a permanent solution at this time.

Bug #2

When attempting to create client connections to a server over localhost, but without internet access, the program will segfault. This is due to how I am doing the hostname lookup, but I did not have time to fix this issue.