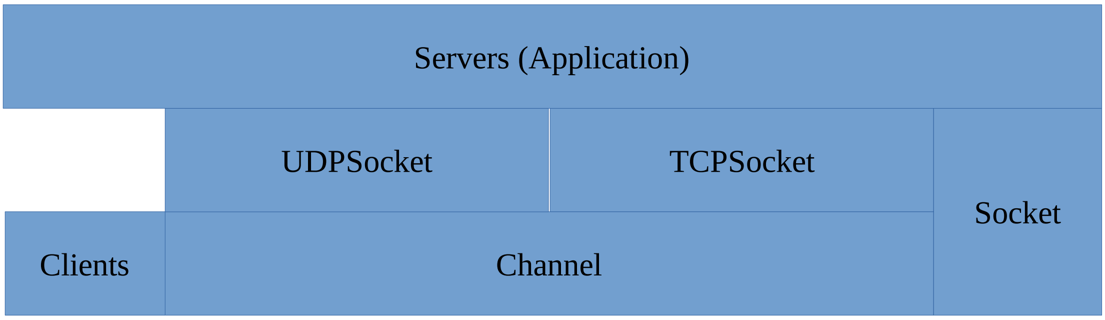


Class structure and design decisions:

The Channel folder in Drivers folder contains the channel.hpp that have abstract class **Channel** that handle socket interactions through its Socket member (which can be TCP or UDP). He have a pointer to a Socket object (Socket* channelSocket) -> (either TCPSocket or UDPSocket). Also have pure virtual functions:

- virtual void start() = 0;
- virtual void stop() = 0;
- virtual void send(const std::string& message) = 0;
- virtual void receive() = 0;

These functions MUST be implemented in derived classes: **serverChannel** and **clientChannel**.

The channel.cpp contains only **serverChannel** functions with the pointer that points to the virtual functions of **Socket**.

The Socket folder in Drivers folder conatins socket.hpp that have abstract class **Socket** that define common socket behaviors (send, receive, connect, shutdown) with pure virtual functions:

- virtual void connect() = 0;
- virtual void send(const std::string& message) = 0;
- virtual void receive() = 0;
- virtual void shutdown() = 0;

These functions MUST be implemented in derived classes: **TCPSocket** and **UDPSocket**.

tcpsocket.cpp and udpsocket.cpp contains the socket behaviour.

UNICAST_TCPSocket:
the client.cpp contains the clientChannel functions with the client behaviour.

MULTICAST_UDPSocket:
the client.cpp contains the clientChannel functions with the client behaviour.

The Server contains the application behaviour: I made a Time class with a += operator overloading so I can increment the time by 1 and wait 1sec and send the time to clients each 1sec.

Note: To run the application, navigate to either the UNICAST_TCPSocket or MULTICAST_UDPSocket folders and execute the make command in the terminal. This will create an out directory containing the server and client executables, which you can easily run.

Step-by-step guide for setting up the development environment (QEMU) with Yocto:

I made an image with tcp-client and udp-server to try each server and client on Qemu.

For the tcp-client:
Going to: **~/yocto/poky/meta/recipes-example/tcp-client** with this tree:

```
├── files
│   ├── CMakeLists.txt
│   ├── Drivers
│   │   ├── Channel
│   │   │   ├── channel.cpp
│   │   │   └── channel.hpp
│   │   └── Socket
│   │       ├── socket.hpp
│   │       ├── tcpsocket.cpp
│   │       └── udpsocket.cpp
│   ├── include
│   │   └── client.hpp
│   └── src
│       ├── client.cpp
│       └── main.cpp
└── tcp-client_0.1.bb
```

For the udp-server:
Going to: **~/yocto/poky/meta/recipes-example/udp-server** with this tree:

```
├── files
│   ├── CMakeLists.txt
│   ├── Drivers
│   │   ├── Channel
│   │   │   ├── channel.cpp
│   │   │   └── channel.hpp
│   │   └── Socket
│   │       ├── socket.hpp
│   │       ├── tcpsocket.cpp
│   │       └── udpsocket.cpp
│   ├── include
│   │   └── server.hpp
│   └── src
│       ├── main.cpp
│       └── server.cpp
└── udp-server_0.1.bb
```

- you will find the recipes and CmakeLists uploaded, take a look on them.

Then going to: **~/yocto/poky/build/conf** and open **local.conf** and add this line:

IMAGE_INSTALL:append = " tcp-client udp-server"

Select:

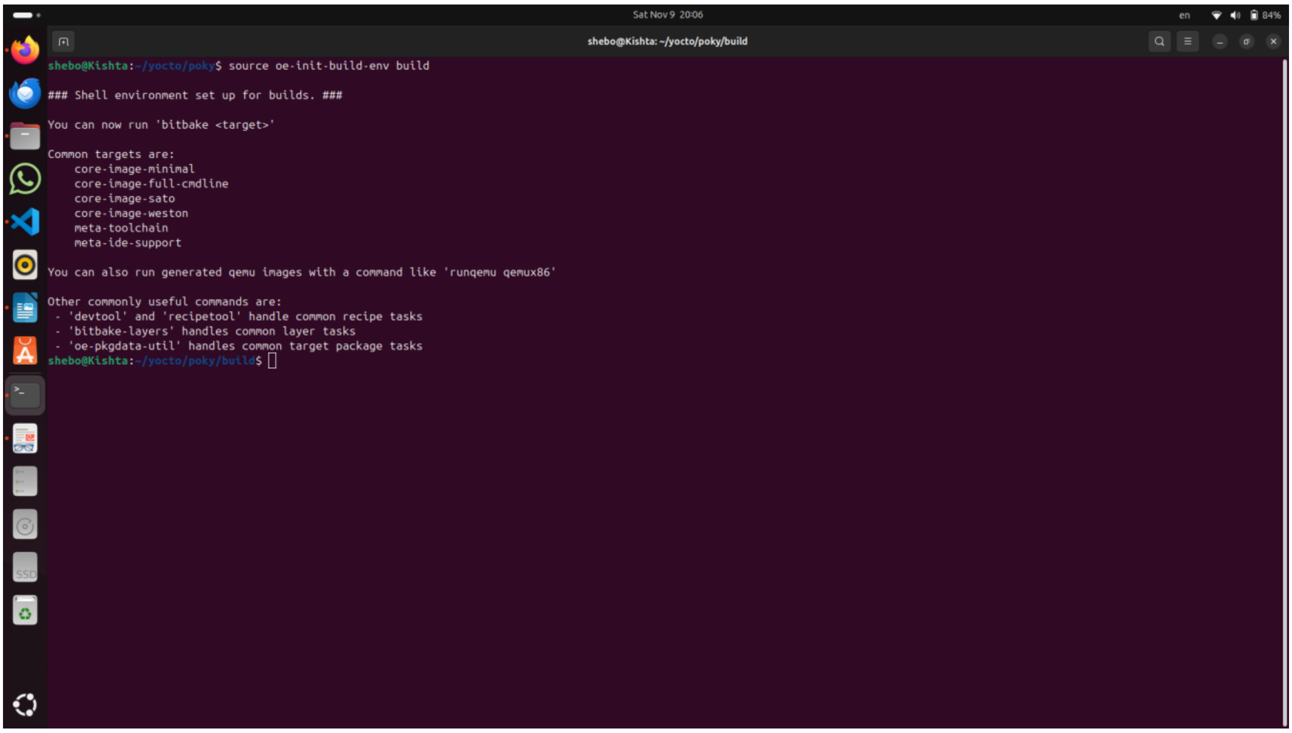
MACHINE ?= "qemuarm64"

to create an image with this applications by yocto, write these commands:

Note:
*-Take the **recipes-example** folder that I uploaded and place it on **~/yocto/poky/meta***

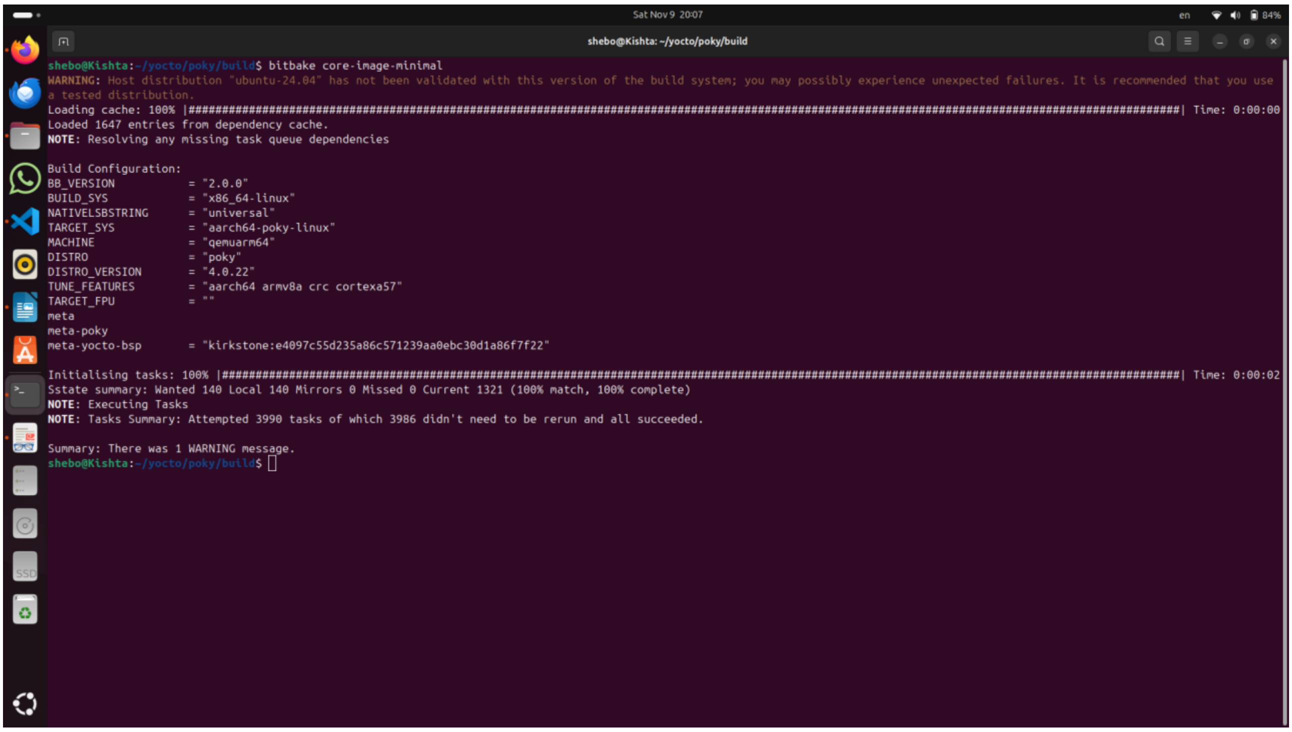
go to: **~/yocto/poky/**

source oe-init-build-env build



```
shebo@Kishta:~/yocto/poky$ source oe-init-build-env build

### Shell environment set up for builds. ###

You can now run 'bitbake <target>'

Common targets are:
    core-image-minimal
    core-image-full-cmdline
    core-image-sato
    core-image-weston
    meta-toolchain
    meta-ide-support

You can also run generated qemu images with a command like 'runqemu qemux86'

Other commonly useful commands are:
 - 'devtool' and 'recipetool' handle common recipe tasks
 - 'bitbake-layers' handles common layer tasks
 - 'oe-pkgdata-util' handles common target package tasks
shebo@Kishta:~/yocto/poky/build$ 
```

bitbake core-image-minimal



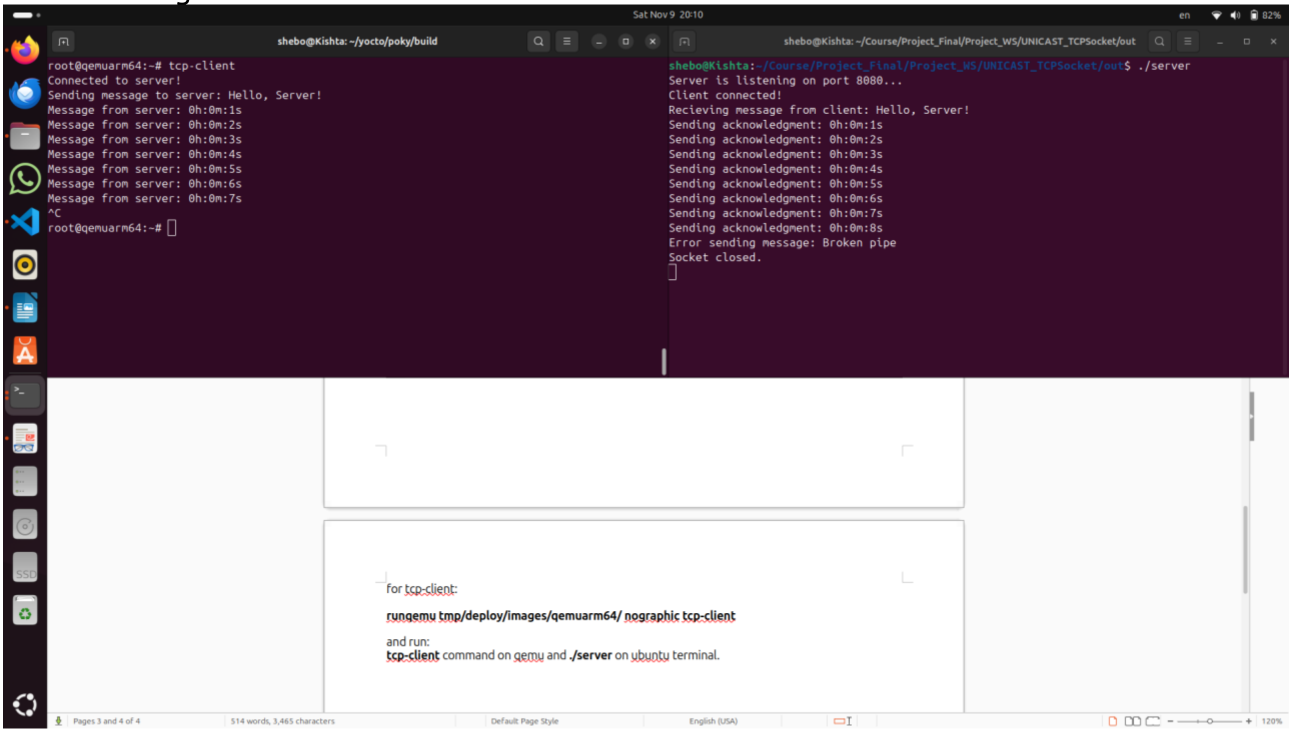
```
shebo@Kishta:~/yocto/poky/build$ bitbake core-image-minimal
WARNING: Host distribution "ubuntu-24.04" has not been validated with this version of the build system; you may possibly experience unexpected failures. It is recommended that you use
a tested distribution.
Loading cache: 100% |#####################################################################################################################################################| Time: 0:00:00
Loaded 1647 entries from dependency cache.
NOTE: Resolving any missing task queue dependencies

Build Configuration:
BB_VERSION           = "2.0.0"
BUILD_SYS            = "x86_64-linux"
NATIVELSBSTRING      = "universal"
TARGET_SYS           = "aarch64-poky-linux"
MACHINE              = "qemuarm64"
DISTRO               = "poky"
DISTRO_VERSION       = "4.0.22"
TUNE_FEATURES        = "aarch64 armv8a crc cortexa57"
TARGET_FPU           = ""
meta
meta-poky
meta-yocto-bsp       = "kirkstone:e4097c55d235a86c571239aa0ebc30d1a86f7f22"

Initialising tasks: 100% |################################################################################################################################################| Time: 0:00:02
Sstate summary: Wanted 140 Local 140 Mirrors 0 Missed 0 Current 1321 (100% match, 100% complete)
NOTE: Executing Tasks
NOTE: Tasks Summary: Attempted 3990 tasks of which 3986 didn't need to be rerun and all succeeded.

Summary: There was 1 WARNING message.
shebo@Kishta:~/yocto/poky/build$ 
```

for tcp-client:

runqemu tmp/deploy/images/qemuarm64/ nographic tcp-client

and run:
tcp-client command on qemu and **./server** on ubuntu terminal.

Disconnecting the client:


Sat Nov 9 20:10
en
82%
shebo@Kishta: ~/yocto/poky/build
shebo@Kishta: ~/Course/Project_Final/Project_WS/UNICAST_TCPSocket/out
Pages 3 and 4 of 4
514 words, 3,465 characters
Default Page Style
English (USA)
120%

```
root@qemuarm64:~# tcp-client
Connected to server!
Sending message to server: Hello, Server!
Message from server: 0h:0m:1s
Message from server: 0h:0m:2s
Message from server: 0h:0m:3s
Message from server: 0h:0m:4s
Message from server: 0h:0m:5s
Message from server: 0h:0m:6s
Message from server: 0h:0m:7s
^C
root@qemuarm64:~# 
```

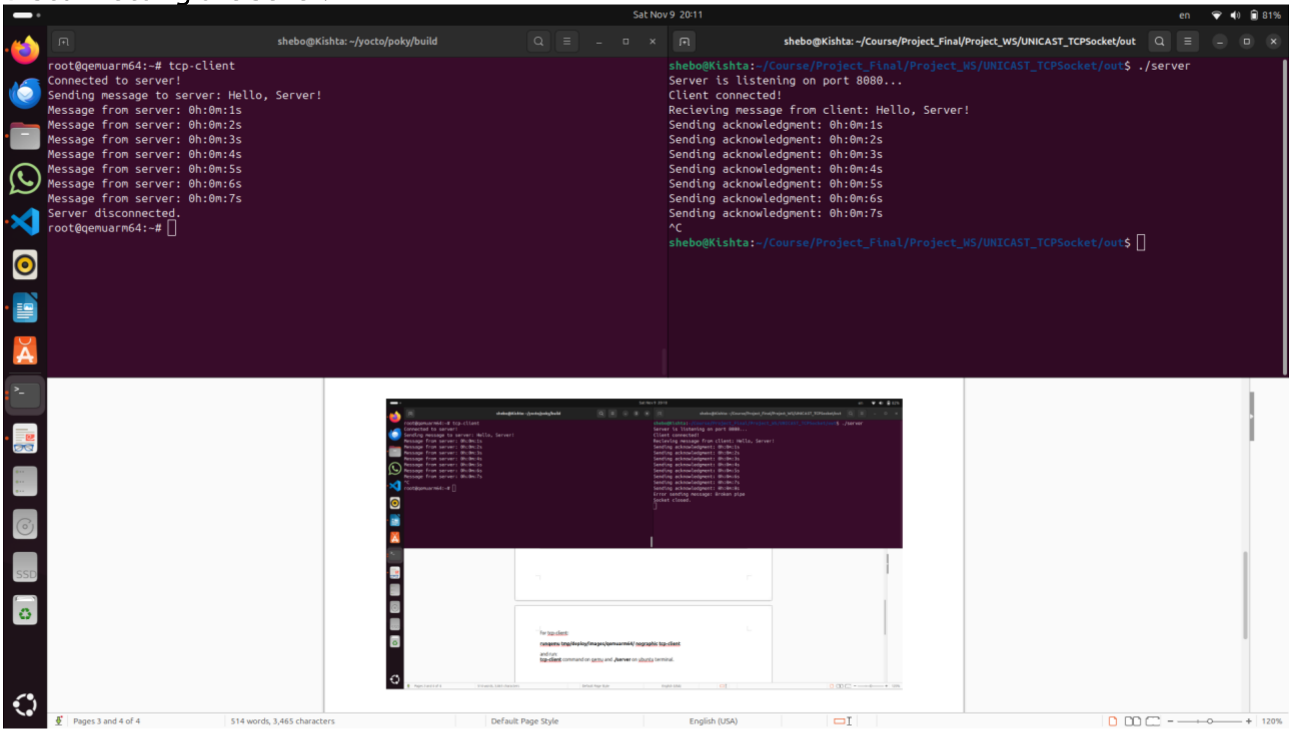
```
shebo@Kishta:~/Course/Project_Final/Project_WS/UNICAST_TCPSocket/out$ ./server
Server is listening on port 8080...
Client connected!
Recieving message from client: Hello, Server!
Sending acknowledgment: 0h:0m:1s
Sending acknowledgment: 0h:0m:2s
Sending acknowledgment: 0h:0m:3s
Sending acknowledgment: 0h:0m:4s
Sending acknowledgment: 0h:0m:5s
Sending acknowledgment: 0h:0m:6s
Sending acknowledgment: 0h:0m:7s
Sending acknowledgment: 0h:0m:8s
Error sending message: Broken pipe
Socket closed.
```

For tcp-client:

runqemu tmp/deploy/images/qemuarm64/ nographic tcp-client

and run:
tcp-client command on qemu and **./server** on ubuntu terminal.

Disconnecting the sever:


Sat Nov 9 20:11
en
81%
shebo@Kishta: ~/yocto/poky/build
shebo@Kishta: ~/Course/Project_Final/Project_WS/UNICAST_TCPSocket/out
Pages 3 and 4 of 4
514 words, 3,465 characters
Default Page Style
English (USA)
120%

```
root@qemuarm64:~# tcp-client
Connected to server!
Sending message to server: Hello, Server!
Message from server: 0h:0m:1s
Message from server: 0h:0m:2s
Message from server: 0h:0m:3s
Message from server: 0h:0m:4s
Message from server: 0h:0m:5s
Message from server: 0h:0m:6s
Message from server: 0h:0m:7s
Server disconnected.
root@qemuarm64:~# 
```

```
shebo@Kishta:~/Course/Project_Final/Project_WS/UNICAST_TCPSocket/out$ ./server
Server is listening on port 8080...
Client connected!
Recieving message from client: Hello, Server!
Sending acknowledgment: 0h:0m:1s
Sending acknowledgment: 0h:0m:2s
Sending acknowledgment: 0h:0m:3s
Sending acknowledgment: 0h:0m:4s
Sending acknowledgment: 0h:0m:5s
Sending acknowledgment: 0h:0m:6s
Sending acknowledgment: 0h:0m:7s
^C
shebo@Kishta:~/Course/Project_Final/Project_WS/UNICAST_TCPSocket/out$ 
```

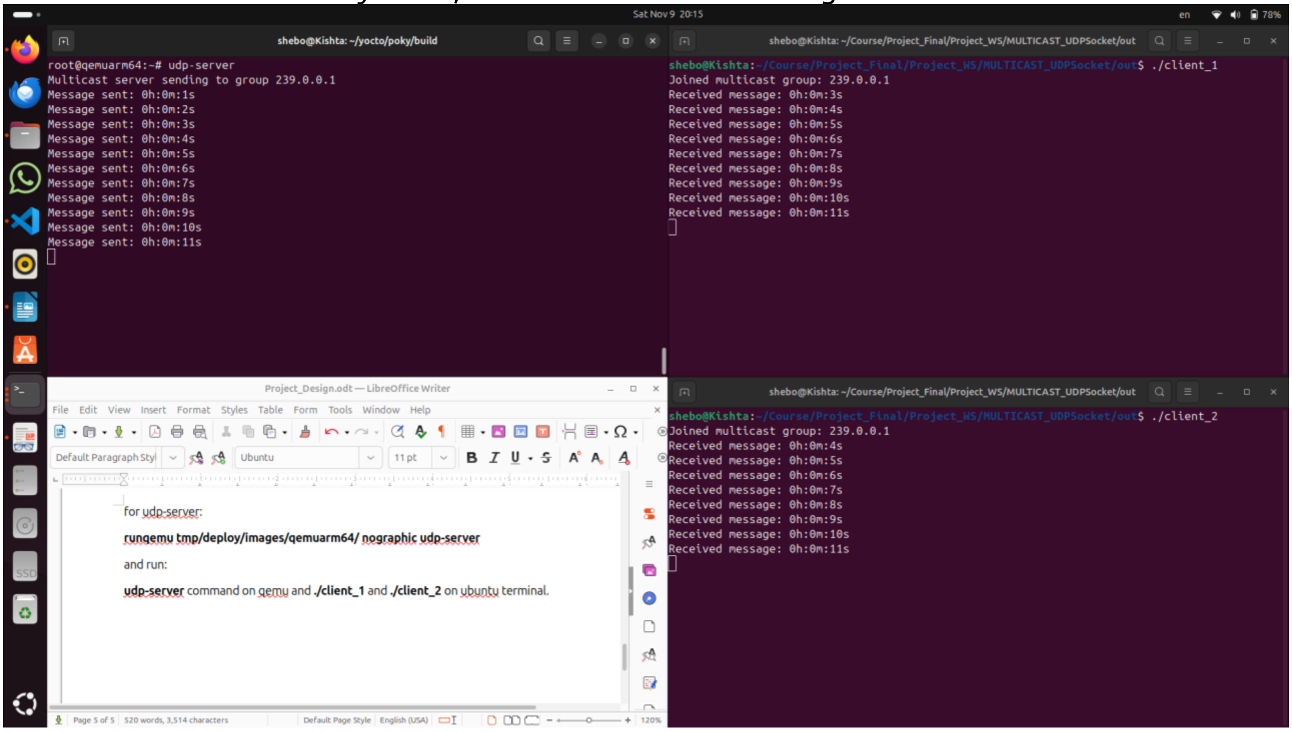
for udp-server:

runqemu tmp/deploy/images/qemuarm64/ nographic udp-server

and run:

udp-server command on qemu and **./client_1** and **./client_2** on ubuntu terminal.

No matters I disconnect any client, the server continue sending the time.



Sat Nov 9 20:15
en
78%
shebo@Kishta: ~/yocto/poky/build
shebo@Kishta: ~/Course/Project_Final/Project_WS/MULTICAST_UDPSocket/out
Project_Design.odt — LibreOffice Writer
File Edit View Insert Format Styles Table Form Tools Window Help
Default Paragraph Styl
Ubuntu
11 pt
Page 5 of 5
520 words, 3,514 characters
Default Page Style
English (USA)
120%

```
root@qemuarm64:~# udp-server
Multicast server sending to group 239.0.0.1
Message sent: 0h:0m:1s
Message sent: 0h:0m:2s
Message sent: 0h:0m:3s
Message sent: 0h:0m:4s
Message sent: 0h:0m:5s
Message sent: 0h:0m:6s
Message sent: 0h:0m:7s
Message sent: 0h:0m:8s
Message sent: 0h:0m:9s
Message sent: 0h:0m:10s
Message sent: 0h:0m:11s
```

```
shebo@Kishta:~/Course/Project_Final/Project_WS/MULTICAST_UDPSocket/out$ ./client_1
Joined multicast group: 239.0.0.1
Received message: 0h:0m:3s
Received message: 0h:0m:4s
Received message: 0h:0m:5s
Received message: 0h:0m:6s
Received message: 0h:0m:7s
Received message: 0h:0m:8s
Received message: 0h:0m:9s
Received message: 0h:0m:10s
Received message: 0h:0m:11s
```

```
shebo@Kishta:~/Course/Project_Final/Project_WS/MULTICAST_UDPSocket/out$ ./client_2
Joined multicast group: 239.0.0.1
Received message: 0h:0m:4s
Received message: 0h:0m:5s
Received message: 0h:0m:6s
Received message: 0h:0m:7s
Received message: 0h:0m:8s
Received message: 0h:0m:9s
Received message: 0h:0m:10s
Received message: 0h:0m:11s
```

for udp-server:

runqemu tmp/deploy/images/qemuarm64/ nographic udp-server

and run:

udp-server command on qemu and **./client_1** and **./client_2** on ubuntu terminal.