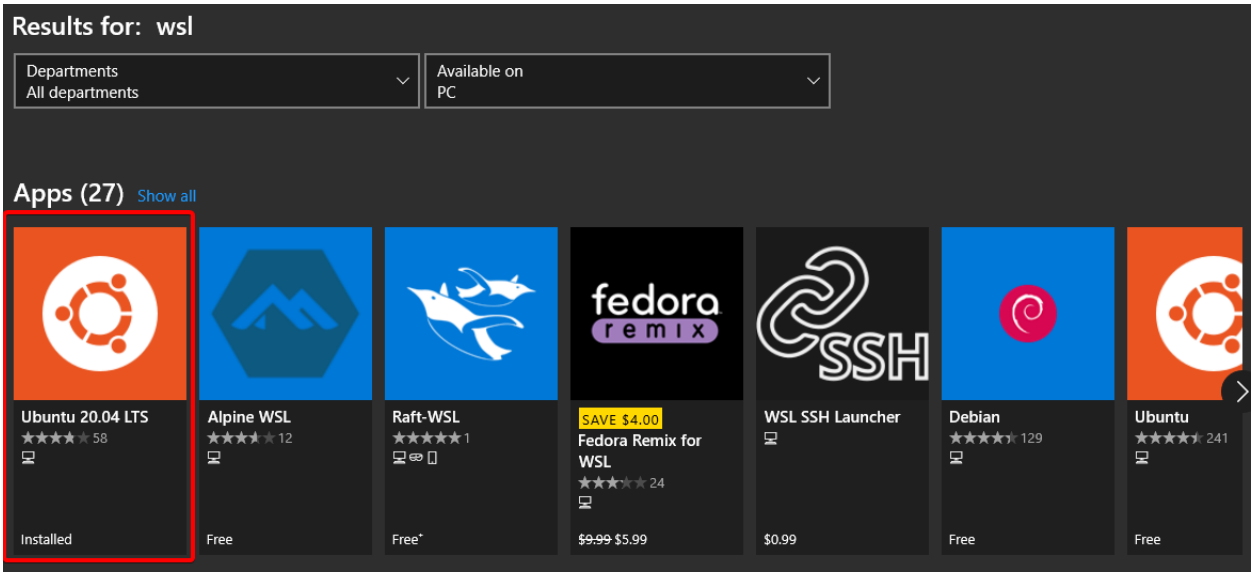
Simulator Setup

Table of Contents

1. Installing Linux for Windows 10 (WSL)[1]

a. Follow steps 1 to 6 in this article: https://docs.microsoft.com/en-us/windows/wsl/install-win10 (These steps may require multiple PC reboots)

b. For step 6, install the latest version of Ubuntu LTS from the Windows store. (currently 20.04)



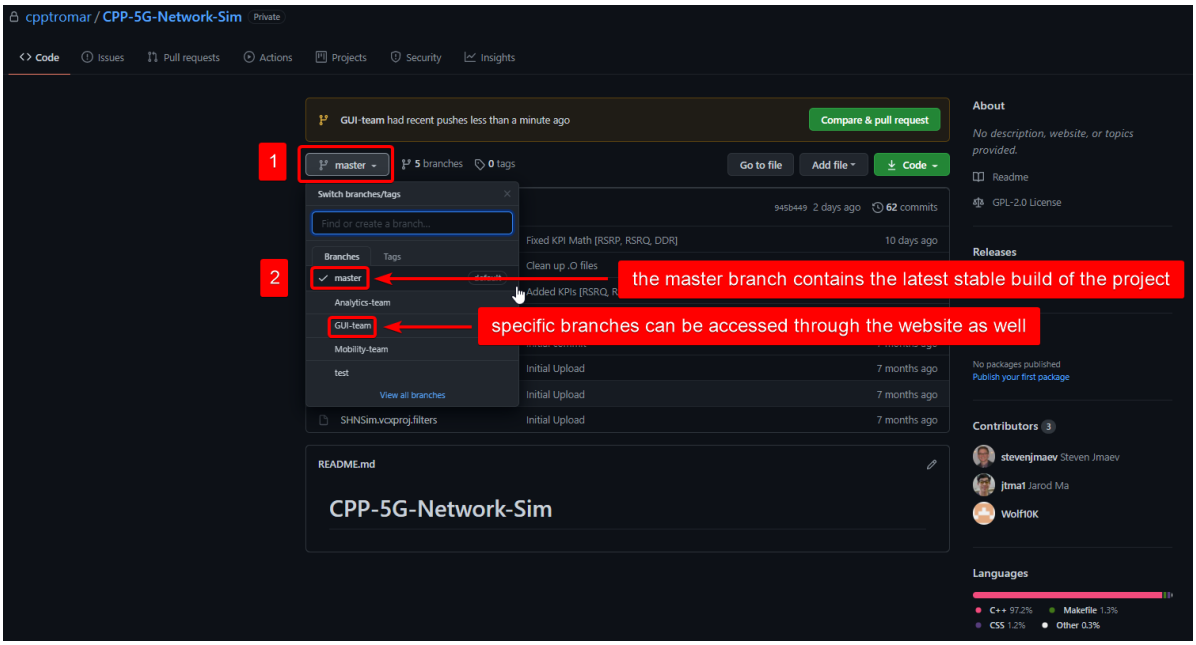
c. After successfully installing Ubuntu 20.04 LTS, open the program through the app store. It will prompt the user to create a password. This password will be used in a future step so be sure to write it down somewhere.

[1] Linux machine or Mac, this step can be skipped.

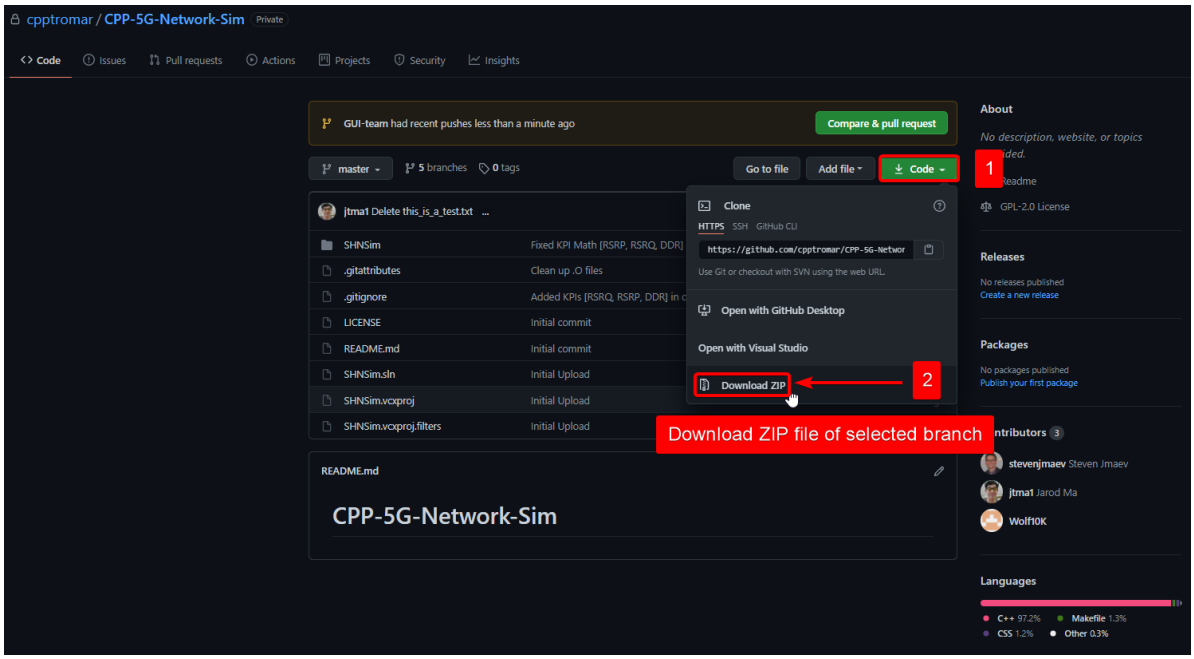
2. Download project files from GitHub[2]

https://github.com/cpptromar/CPP-5G-Network-Sim[3]

a. Choose version to download



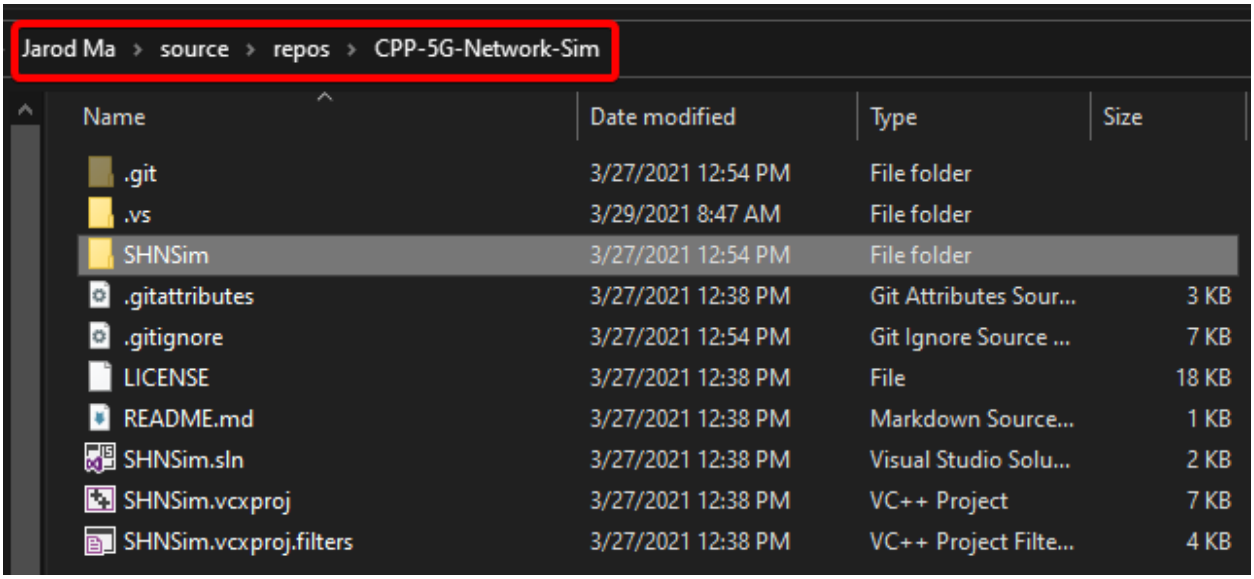
b. Download ZIP file



[2] A basic tutorial related to how GitHub works can be found here: https://youtu.be/w3jLJU7DT5E

[3] This link requires permission to access the files and may change in the future depending on the owner of the repo.

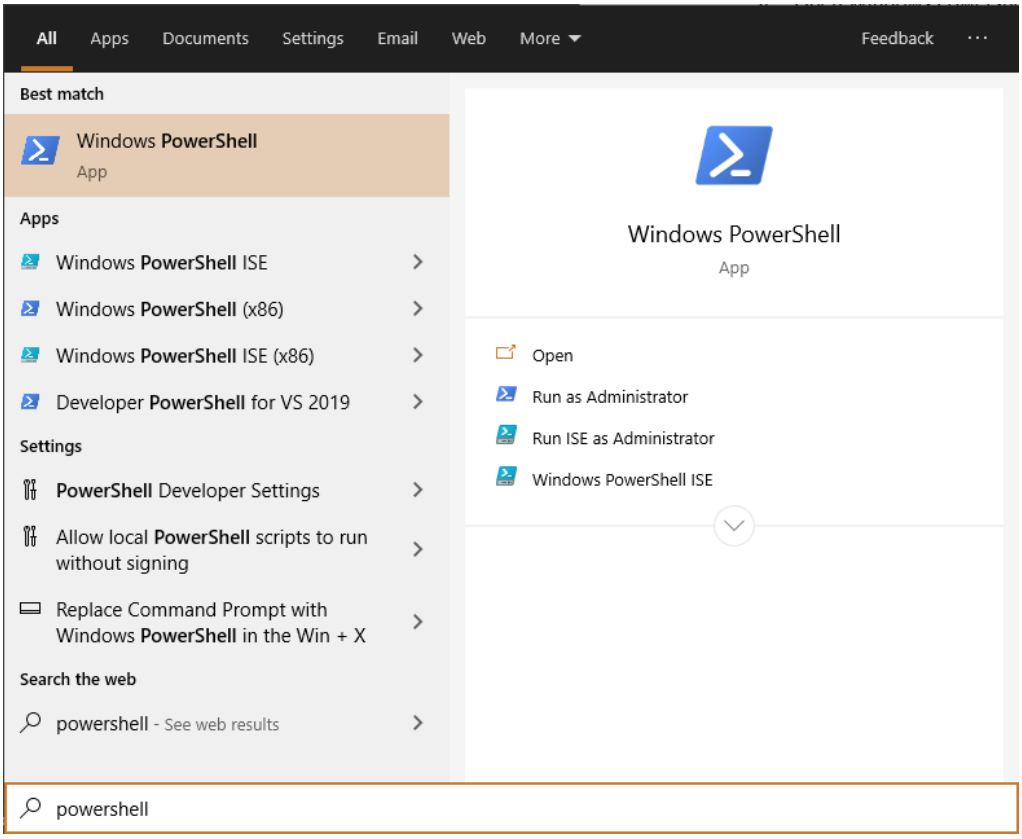
c. Extract ZIP file to a folder[4]



[4] We will be running all the Linux commands within the SHNSim folder. Be sure to extract the ZIP files to a folder instead of attempting to run the commands within the ZIP.

3. Installing libraries for WSL

a. Open Windows PowerShell[5]



b. Change directory to wherever the SHNSim folder is located

```
cd [path\to\SHNSim_folder]
```

c. Run WSL

```
wsl
```

d. Install all required packages by running the following commands

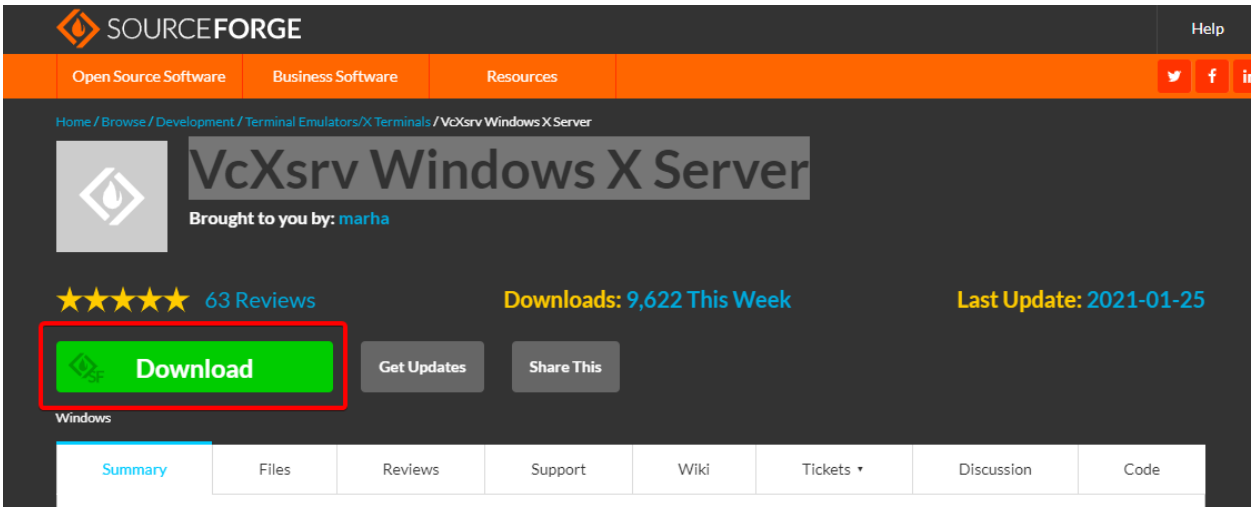
GTK & Dependency Libraries (for GUI)
sudo apt-get update sudo apt-get install libgtk-3-dev sudo apt-get install libgtk-3-0 sudo apt-get install pkg-config sudo apt-get install libcairo2-dev
Makefile & G++ Libraries (for compiling)
sudo apt install make sudo apt install make-guile sudo apt install g++

[5] For Mac/Linux/Ubuntu, use the terminal application to run command lines. The “wsl” command does not need to be executed because it is used to start Windows Subsystem for Linux (WSL).

4. Installing VcXsrv Windows X Server[6]

a. Download VcXsrv from their official website and run their installer (may require a PC reboot)

https://sourceforge.net/projects/vcxsrv/



b. Run this command in PowerShell

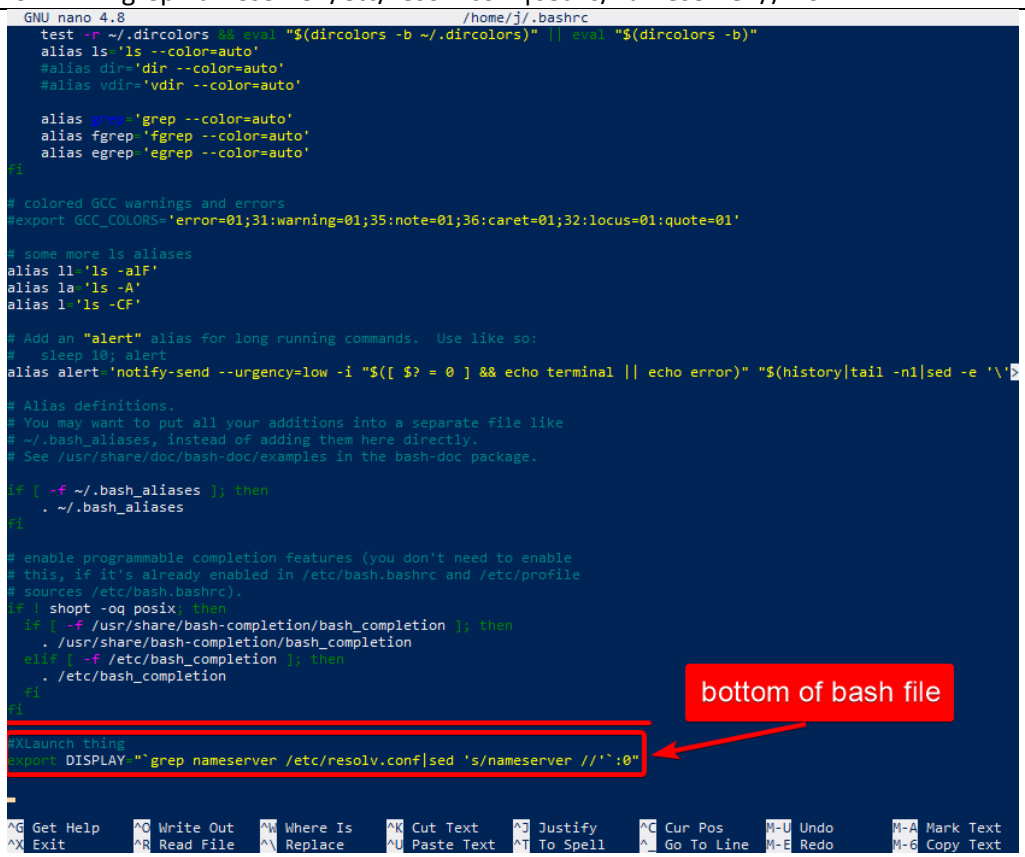
```
sudo nano ~/.bashrc
```

c. Enter your password that you set earlier when running WSL for the first time.

[6] This application is used as a virtual display when running the "./output" command. Mac/Linux/Ubuntu users may require a different program to support displaying the simulator. Refer to the C++ GTK documentation for more information (https://www.gtk.org/docs/language-bindings/cpp).

d. Use the down arrow key to navigate to the bottom of the bash file and add the following code:

```
export DISPLAY="`grep nameserver /etc/resolv.conf|sed 's/nameserver //'`:0"
```

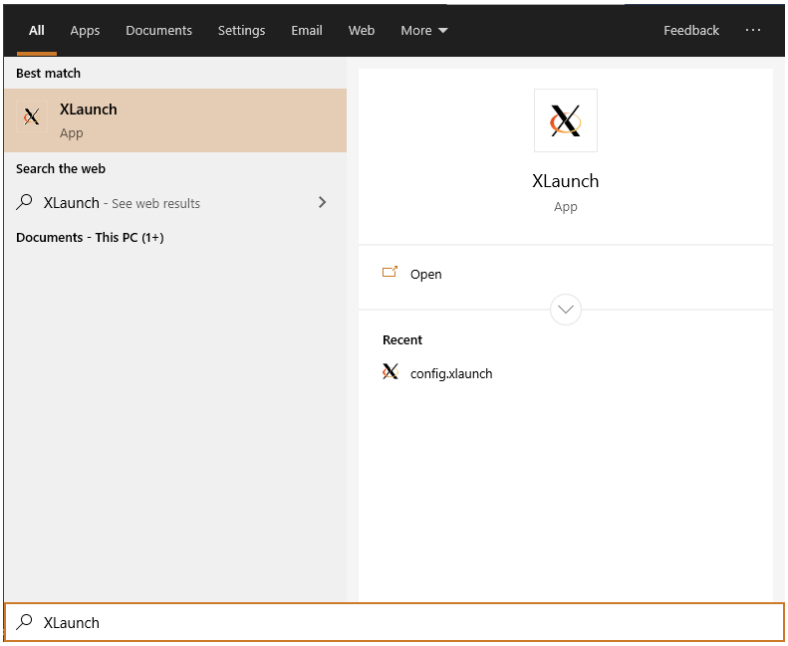


e. Save and close the bash file using these three key combinations in order: CTRL+O, Enter, CTRL+X

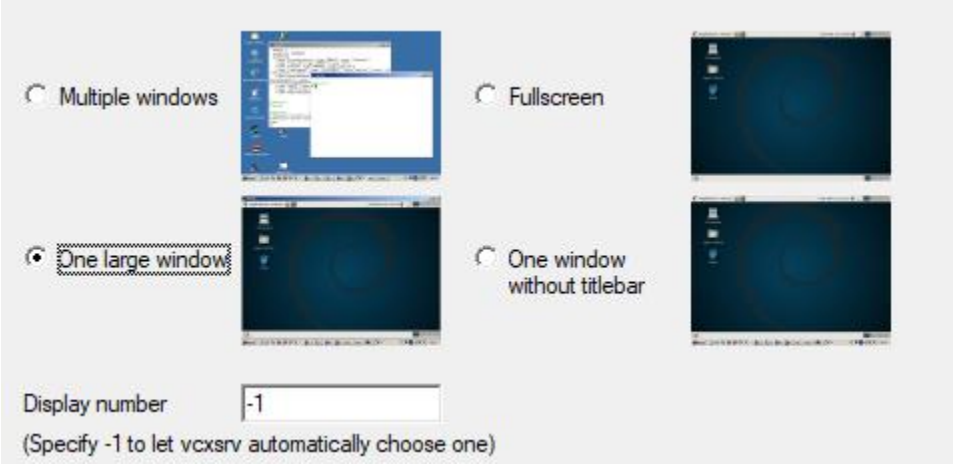
f. Close and re-open PowerShell

5. Configuring XLaunch[7]

a. Open XLaunch through windows explorer

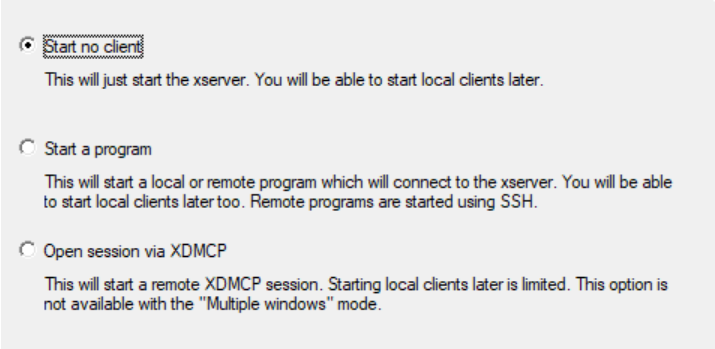


b. Select “One Large Window” and then press “Next”

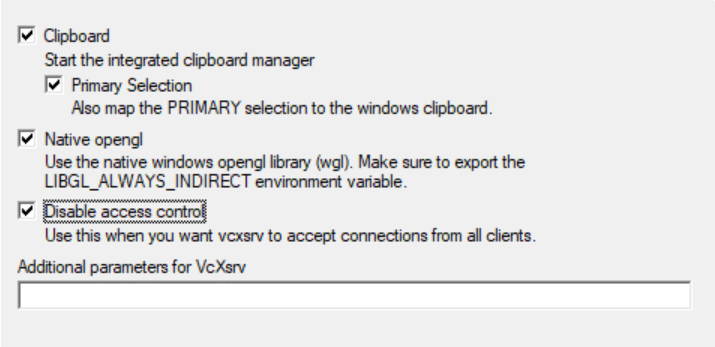


[7] The most efficient way to use XLaunch is to create a .config file that will be opened every time the simulator is opened. These following steps will show the process of generating a .config file using the XLaunch setup wizard.

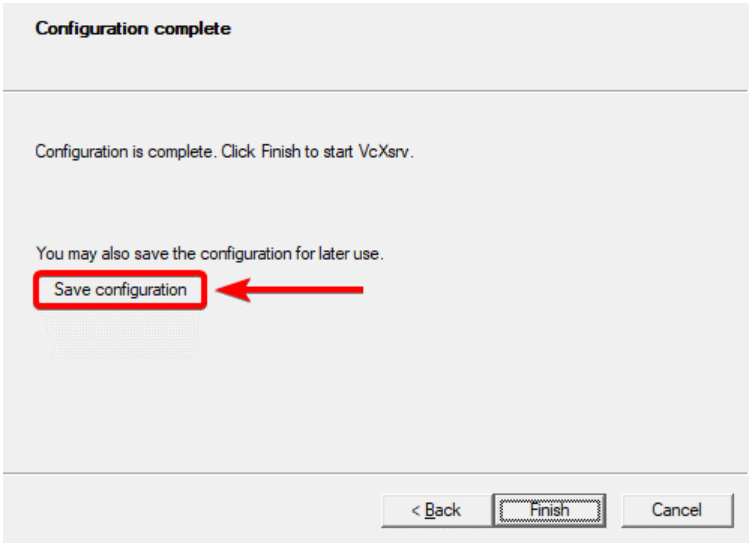
c. Select “Start no client” and then press “Next”

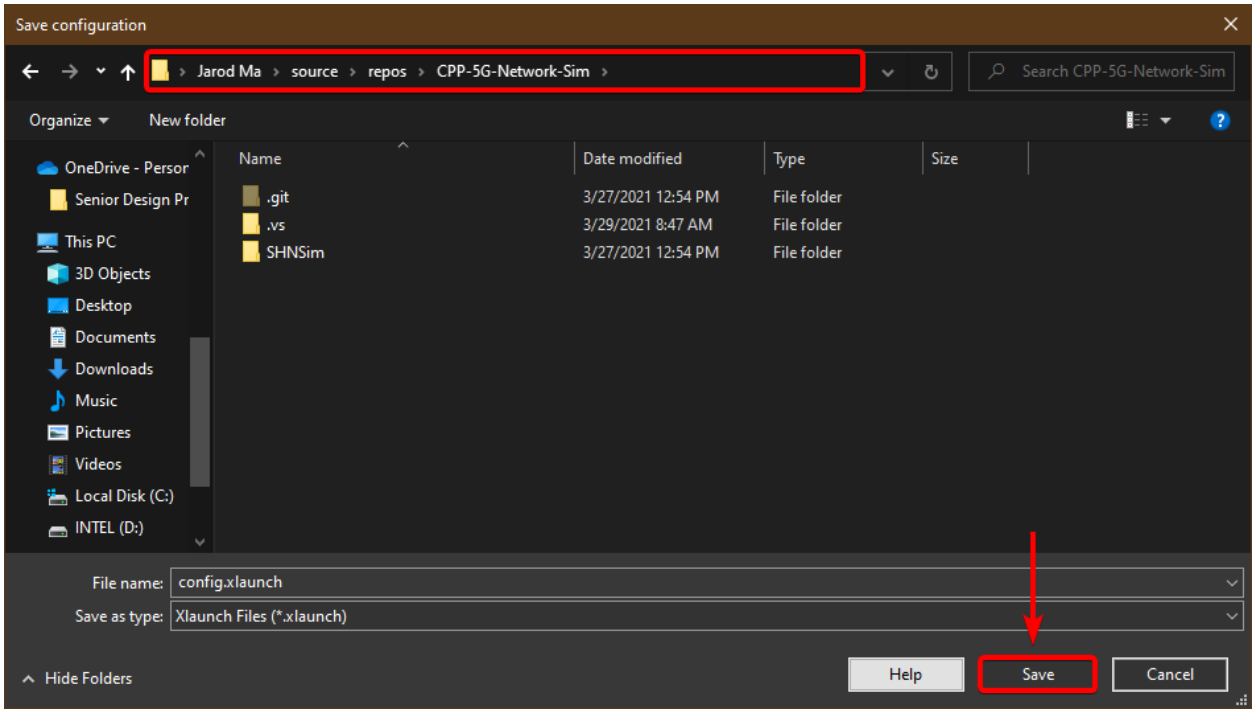


d. Check “Disable Access Control” and then press “Next”



e. Save configuration (.config file) to somewhere accessible. (e.g. Desktop or Project location)




Save configuration
Jarod Ma
source
repos
CPP-5G-Network-Sim
Search CPP-5G-Network-Sim
Organize
New folder
OneDrive - Persor
Senior Design Pr
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
INTEL (D:)
Name
Date modified
Type
Size
.git
3/27/2021 12:54 PM
File folder
.vs
3/29/2021 8:47 AM
File folder
SHNSim
3/27/2021 12:54 PM
File folder
File name:
config.xlaunch
Save as type:
Xlaunch Files (*.xlaunch)
Hide Folders
Help
Save
Cancel

6. Compiling the code with makefile and running the simulator[8]

a. Open XLaunch through the configuration file. This will open a blank window.
b. Change directory to wherever the SHNSim folder is located

```
cd [path\to\SHNSim_folder]
```

c. Run WSL

```
wsl
```

d. Compile the code with makefile

```
make output
make clean
```

e. Output to Display

```
./output
```

[8] These steps should be executed every time the simulator's code is modified. To run the simulation without compiling with makefile: Opening Xserver, changing the directory, running WSL, and then using "./output" is sufficient.

7. Running an executable on a Linux machine

Method 1:

a. In the terminal, change directory to wherever the SHNSim folder is located.

```
cd [path\to\SHNSim_folder]
```

b. Update permissions for the built-in executable.

```
chmod u+x compile.sh
```

c. Run the simulator

```
./compile.sh
```

Method 2:

a. Change directory on the system to wherever the SHNSim folder is located.

```
cd [path\to\SHNSim_folder]
```

b. Within the directory, create a new script file containing the following commands:

```
#!/bin/bash
make output
make clean
./output
```

c. Change permissions by running chmod

```
chmod u+x <filename>
```

d. Run the script

```
./<filename>
```

e. Once the script has been run once, change the script's directory to an easily accessible location such as the desktop.

f. The script should now be able to be run from double clicking the file.

*This has only been tested on a virtual machine running ubuntu 20.04.