**Christmas Kiosk rPi Configuration**

1. Use Raspberry Pi Imager to load the latest OS to your Pi (GUI version, not headless)
   1. Ensure you set your WiFi SSID and credentials when flashing the rPi so that you can SSH (Putty) into it after
   2. Key difference in latest Raspian is the new usage of labwc which changes the screen and touchpad rotations approaches – the good news is that it’s much easier now than shown in older tutorials
2. SSH (Putty) into Pi to Update all packages
   1. **sudo apt update**
   2. **sudo apt full-upgrade**
3. Reboot Pi and then SSH back in
   1. **sudo reboot**
4. Configure ElecLab touchscreen
   1. **sudo nano /boot/firmware/config.txt**
   2. Add the following lines (if using a different touchscreen, these commands may be different.
      1. **hdmi\_force\_hotplug=1**
      2. **hdmi\_mode=87**
      3. **hdmi\_ignore\_edid=0xa5000080**
      4. **hdmi\_cvt 1280 800 60 3 0 0 0**
      5. **hdmi\_group=2**
      6. **hdmi\_drive=2**
   3. Save and exit the editor (typically Control+O then Control+X)
   4. You can also download the completed config.txt file here
      1. <https://drive.google.com/drive/folders/1NAklSN3KqI5Ndo3yUlYqIJBJxxRZP8VJ>
5. Use command line raspi-config to change display driver
   1. **sudo raspi-config**
   2. Make the following changes in the text-based GUI that opens
      1. *6. Advanced Options ->A6 Wayland (Switch between x and Wayland backends) -> W3 Labwc*
         1. This will allow us to change the display orientation through the OS GUI rather than a bunch of command line configurations that can vary from system to system
      2. *Ok, finish, reboot*
6. (Optional - Could also do this through the touchscreen itself) Enable VNC Server
   1. We need to access the GUI to change the display orientation in Labwc
   2. **sudo raspi-config**
   3. *Interface Options -> VNC -> Would you like VNC Server Enabled -> Yes*
   4. You can now use TigerVNC on your host machine to access the desktop if that is preferred to using the touchscreen
7. Change display orientation in OS GUI
   1. NOTE: Touchscreen must be installed before completing this step or you will not see the monitor in screen preferences
   2. If you are using the same touch screen I am and want a portrait (tall) orientation:
      1. *OS GUI: Pi menu -> Preferences -> Screen Configuration*
         1. *“Screens” button (bottom left)*
         2. Select your screen (something like HDMI-A-1) from the menu that opens
         3. Select Orientation and then set it as defined below
            1. Connectors down: Select *Right*
            2. Connectors up: Select *Left*
         4. *Click Apply, Then OK* on the confirmation window that pops up (if everything looks good)
         5. Close the window
8. (Optional - Only needed if you plan on using the GPIO pins too) Add web server, PHP, and SQL
   1. Reference:<https://randomnerdtutorials.com/raspberry-pi-apache-mysql-php-lamp-server/>
   2. Commands:
      1. **sudo apt-get install php libapache2-mod-php**
      2. **sudo apt install mariadb-server php-mysql -y**
      3. **sudo service apache2 restart**
      4. **sudo mysql\_secure\_installation**
   3. Give pi user ownership of Apache directories
      1. **sudo chown -R pi /var/www/**
9. (Optional) Add GPIO control to webserver (if adding GPIO pin control functionality to local web server - e.g. speaker relay control)
   1. References:
      1. <https://raspberrypi.stackexchange.com/questions/135673/can-you-control-raspberry-pi-gpio-pins-from-a-html-website>
      2. <https://raspberrypi.stackexchange.com/questions/39139/webserver-gpio-python-script-no-access-to-dev-mem-try-running-as-root/39191#39191>
   2. Give GPIO permissions to www data (Apache)
      1. **sudo groupadd gpio**
      2. **sudo usermod -a -G gpio www-data**
      3. **sudo grep gpio /etc/group**
      4. **sudo chown root:gpio /dev/gpiomem**
      5. **sudo chmod g+rw /dev/gpiomem**
      6. **sudo chmod 776 /home/pi**
      7. **sudo usermod -a -G pi www-data**
10. Create a script to open chromium in kiosk mode
    1. Reference: <https://forums.raspberrypi.com/viewtopic.php?t=378883>
    2. Commands:
       1. **cd /home/pi**
       2. **sudo nano kiosk\_refresh**
          1. This is the file you will use to tune your chromium kiosk settings going forward
       3. Add the following lines to the end of the file and save it.
          1. *#!/bin/sh*
          2. *chromium = /usr/bin/chromium-browser --start-fullscreen --start-maximized --kiosk --noerrdialogs --disable-default-apps --disable-single-click-autofill --disable-translate-new-ux --disable-translate --disable-cache --disk-cache-size=1 --reduce-security-for-testing --app=*[*http://localhost*](http://localhost)
          3. \*\* REPLACE <http://localhost> with your desired website (e.g. your Remote Falcon link). If hosting your own page, you’ll use localhost, but that has additional steps.
          4. Save and exit the editor (typically Control+O then Control+X)
       4. Make kiosk\_refresh executable
          1. **sudo chmod 777 kiosk\_refresh**
       5. You can also download a kiosk\_refresh script here: <https://drive.google.com/drive/folders/1NAklSN3KqI5Ndo3yUlYqIJBJxxRZP8VJ>
          1. Simply download it and replace my Remote Falcon link with yours
11. (Optional) Add a cron job to reset the browser page periodically in case something happens or a user does something unexpected
    1. **crontab -e**
    2. Say yes if prompted to create a new cron jobs file
       1. This just means you didn’t have any existing cron jobs
    3. Add the following line to the bottom of the text editor that opens
       1. ***\*/5 \* \* \* \* /home/pi/kiosk\_refresh***
          1. Where **5** is how many minutes between refreshes and **kiosk\_refresh** is the script you want to run at that interval
       2. Save and exit the editor (typically Control+O then Control+X)
12. Update autostart commands
    1. **cd /etc/xdg/labwc**
    2. **sudo nano autostart**
    3. Add the following line to the end of the file and save it
       1. **/home/pi/kiosk\_refresh**
       2. Save and exit the editor (typically Control+O then Control+X)
    4. TIP: After doing this you won’t have access to the OS GUI anymore (unless you attach a keyboard/mouse and exit the kiosk). You can use the command below in an SSH session to end the kiosk and return to the OS
       1. **sudo pkill chromium**
13. Set kiosk mode to automatically start on power on
    1. Setup auto login via raspi-config
       1. **sudo raspi-config**
       2. *1 System Options -> S5 Boot / Auto Login -> B4 Desktop Autologin*
       3. *Ok and Reboot*
14. Test It (Let’s see if everything is working so far!)
    1. Reboot the pi and watch the display. You may briefly see the OS load, but then it should immediately load the default Apache web page or your remote falcon site in a kiosk mode browser in the desired orientation. Also move the mouse around on your touch screen to make sure it moves as expected
       1. **sudo reboot**
15. (Optional) Calling php script from button press
    1. I recommend starting with the files located on this share and then modifying them to suit your needs: <https://drive.google.com/drive/folders/1BSQaebGdE7-Egv5zdeDVujFOT_4LATiY>
       1. index.html
       2. LocalActions.php
       3. gpioMomentary.py
    2. Reference:
       1. <https://www.geeksforgeeks.org/how-to-call-php-function-on-the-click-of-a-button/>
16. (Optional) Security and Stability
    1. There are a number of optional configurations you can set to make your system more stable and secure
       1. Turn off Bluetooth
          1. **sudo nano /boot/firmware/config.txt**
          2. add **dtoverlay=disable-bt**
       2. Make SD card read only (will help prevent corruption if pulling power rather than performing proper shutdowns)
          1. **sudo mount -o remount,rw /boot**
          2. A better option would be a cron job to perform shutdown at a specific time that precedes killing power to the kiosk (**sudo shutdown**)