Environment setup for DUT Lab

I. Target:

Bring up WeOS Opensource Edition to Raspberry PI4

- Target system: Raspberry PI 4

- Host system: window x86/Linux x86

- SW platform and framework: WebOS Open Source Edition.

II. Setup WebOS environment

1. Setup Ubuntu local server

a. Prepare Your Ubuntu Environment

Update Your System:

First, ensure that your system is up-to-date:

```
sudo apt update
sudo apt upgrade
```

b. Install Dependencies

Install Essential Tools and Libraries:

You'll need several development tools and libraries. Install them using:

```
  sudo apt install build-essential cmake git curl
```

Install Required Software:

webOS OSE requires specific tools. Install them with:

```
sudo apt install autoconf automake bison flex gawk libtool pkg-config
```

c. Install Docker

webOS OSE uses Docker for containerization. Install Docker by following these steps:

Install Docker:

```
sudo apt install docker.io
```

Enable Docker to Start on Boot:

```
sudo systemctl enable docker
```

Start Docker:

```
sudo systemctl start docker
```

Add Your User to the Docker Group (Optional but recommended):

```
sudo usermod -aG docker $USER
```

You'll need to log out and log back in for this change to take effect.

2.Clone and install WebOS

a. Install git

On Windows 10
https://git-scm.com/download/win

Installing on Linux

```
 sudo dnf install git-all
```

on a Debian-based distribution, such as Ubuntu, try apt:

```
sudo apt install git-all
```

On Mac https://git-scm.com/download/mac

b. Download source code

1. Download source codes

```
git clone https://github.com/webososе/build-webos.git

cd build-webos

git checkout <branch of the latest commit>
```

c. Install and configure the build

a. Install and configure the build

```
sudo scripts/prerequisites.sh

./mcf -p <num of CPUs> -b <num of CPUs> <device type>

*ref: Building webOS OSE | webOS Open Source Edition
```

b. Start to build

```
source oe-init-build-env

bitbake webos-image
```

c. Tips

At step run "*mcf*", we can use "*premirror*" in-case the connection to Git Hub is not smooth.

```
./mcf -p 2 -b 2 --premirror http://webosimg.lge.com/downloads --sstatemirror http://webosimg.lge.com/build-artifacts/webos/master/sstate-cache raspberrypi4-64
```

III. For github environment:

Creating a GitHub environment involves setting up your GitHub repository, configuring your local development environment, and possibly setting up automated workflows. Here's a step-by-step guide to get you started:

1.Sign Up and Create a GitHub Account

- Go to GitHub and sign up for a free account.
- Follow the instructions to verify your email and set up your profile.

2. Configure git

Open your terminal or command prompt and configure Git with your personal details:

```
git config --global user.name "Your Name"
git config --global user.email "your.email@example.com"
```

a. Create a New Repository on GitHub

Go to your GitHub dashboard and click on the "New" button or the "+" icon and select "New repository."

Fill out the repository name, description, and choose between public or private.

Initialize the repository with a README if you want.

b. Clone the Repository Locally

Copy the URL of your new repository from GitHub.

Open your terminal and run:

```
git clone https://github.com/yourusername/your-repository.git
```

Navigate into your repository directory:

```
cd your-repository
```

c. Add and Commit Changes

Make changes to your files in your local repository.

Add changes to the staging area:

```
git add .
```

Commit the changes with a message:

```
git commit -m "Initial commit message"
```

d. Push Changes to GitHub

Push your commits to GitHub:

```
git push origin main
```

If you're working on a different branch, replace main with your branch name.

e. Set Up Branches

To create a new branch:

```
git checkout -b new-branch-name
```

To switch branches:

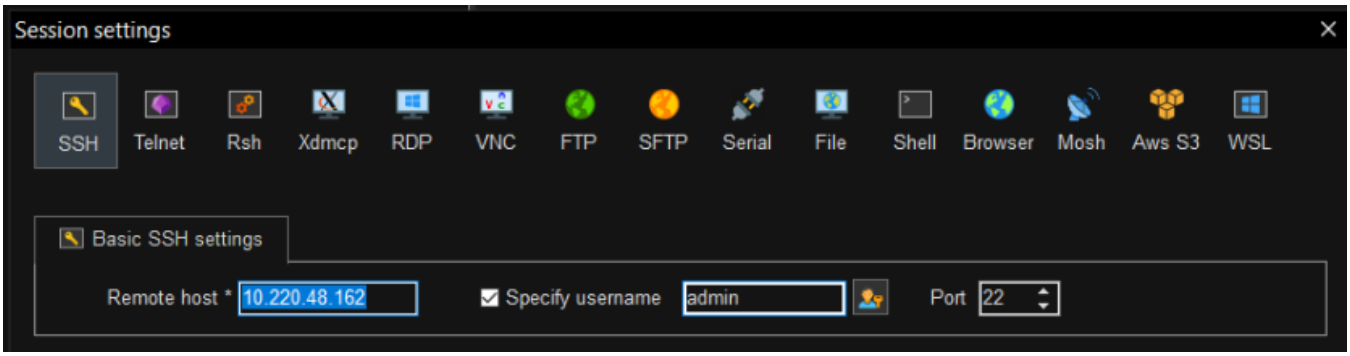
```
git checkout branch-name
```

To merge branches:

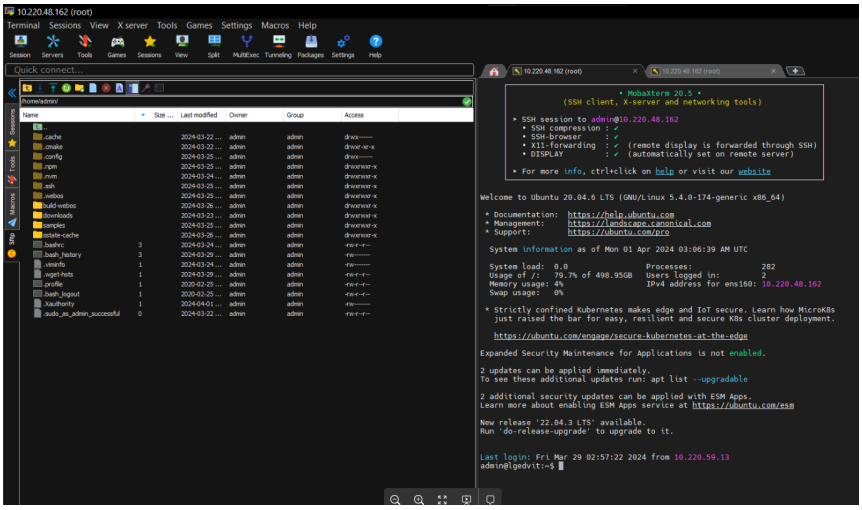
```
git checkout main
git merge new-branch-name
```

IV.Access to local server:

- Use moba Xterm:



- After login:



- Go to directory:

```
cd /home/admin/build-webos/
```

- This is directory you can start to build webOS image:

```
# Start to build

source oe-init-build-env
bitbake webos-image
```

- Since webOS image is built successfully, you can get the image in this directory:

```
cd /home/admin/build-webos/BUILD/deploy/images/raspberrypi4-64/
```

Images:

```
webos-image-devel-raspberrypi4-64.rootfs.wic.bz2

webos-image-raspberrypi4-64.rootfs.wic.bz2
```

- To flash webOS image into board, please download the image from local build then follow the guide in this link: Flashing webOS OSE | webOS Open Source Edition