**Working Guide**

*Anydesk Steps:*

# Install AnyDesk to NUC and Android tablet

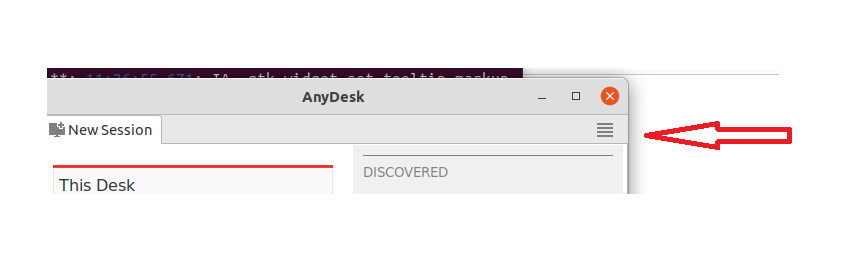
## Install **AnyDesk** on your **Android**

## Install **AnyDesk** On **NUC**

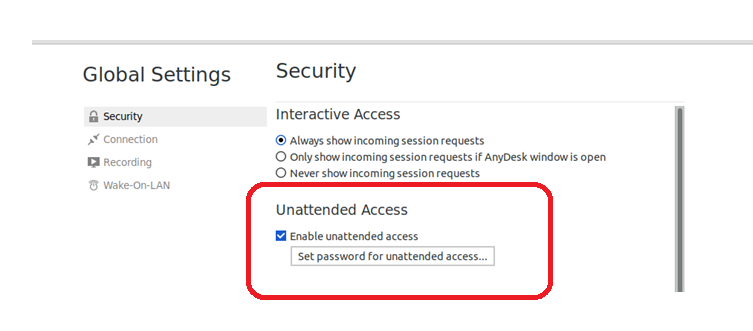
## **( NUC )** on Linux turn on automatic login

## **( NUC )** in Terminal run : **Anydesk**

## Tap on Settings



## (NUC )turn on access with password



## **Anydesk:** Password: (set any password by clicking “set password for unattended access”.

## Create HOT SPOT on **Android** tablet and connect to this HOT SPOT from **NUC**

## OPTIONAL: Restart **NUC** and check connection from **Android** tablet

# Create Remote screen for AnyDesk (restart)

## in Terminal : cd /usr/share/X11/xorg.conf.d/

## create file with name **10-headless.conf**

## insert content to file:

Section "Device"

Identifier "Configured Video Device"

Driver "dummy"

EndSection

Section "Monitor"

Identifier "Configured Monitor"

HorizSync 31.5-48.5

VertRefresh 50-70

EndSection

Section "Screen"

Identifier "Default Screen"

Monitor "Configured Monitor"

Device "Configured Video Device"

DefaultDepth 24

SubSection "Display"

Depth 24

Modes "1920x1080"

EndSubSection

EndSection

## Save file

## In Terminal: sync

## Restart **NUC**. **After restart your screen will be black,** but you can connect with **Android**.

***System User Manual***

**[Bash files directory : guide\_me\_ros/scripts/bash automation/]**

**Return to Physical screen :**

* Activate bash script “phys\_screen.sh”
* restart device

**Return to Remote Screen (using the system remotely) :**

* Activate bash script “rem\_screen.sh”
* restart device

! Make sure you first completed step 1. from “anydesk steps” above !

# Connect with AnyDesk to NUC

* 1. Open HotSpot on **Android** tablet
  2. From Android: open **AnyDesk** and Log In to **NUC**

# Set USB permission for Vibration Device

* Activate bash script “vibration\_usb.sh”

# Start realsense camera

## Activate bash script “cam\_launch.sh”



# Start the algorithm: Open VS CODE and start **main\_multi.py**

**[**path: guide\_me\_ros/Algo/Python/realtime/**]**

***Github Tutorial***

1. **Updating the files locally:**

* **git pull**

1. **Adding local changes to the next commit update**

* **git add .** (the dot at the end is a part of the command)

1. **Creating a commit (snapshot of the current, updated repository state):**

* **git commit -m “write your message regarding the update here”**

1. **Pushing the changes remotely (updating the repository)**

* **git push**

**[**Pay attention - when git requests for username and password, for the password you input the git credential token**].**