

用 Nix 打造 可重現的 Linux 桌面

willie

Outline

- Nix 基礎
- Linux Ricing

<https://devvillie.me/blog/hackersir-nix/>

Nix 是什麼？

Nix: 套件管理器

NixOS: 系統



套件管理器 (Package Manager)

Debian 系: apt

Arch: pacman

Fedora: dnf

NixOS: nix

Nix 與一般套件管理器有何不同？

- Reproducible: 可重現性
- Declarative: 宣告式
- Reliable: 可靠

Reproducible, Declarative

- Nix 語言

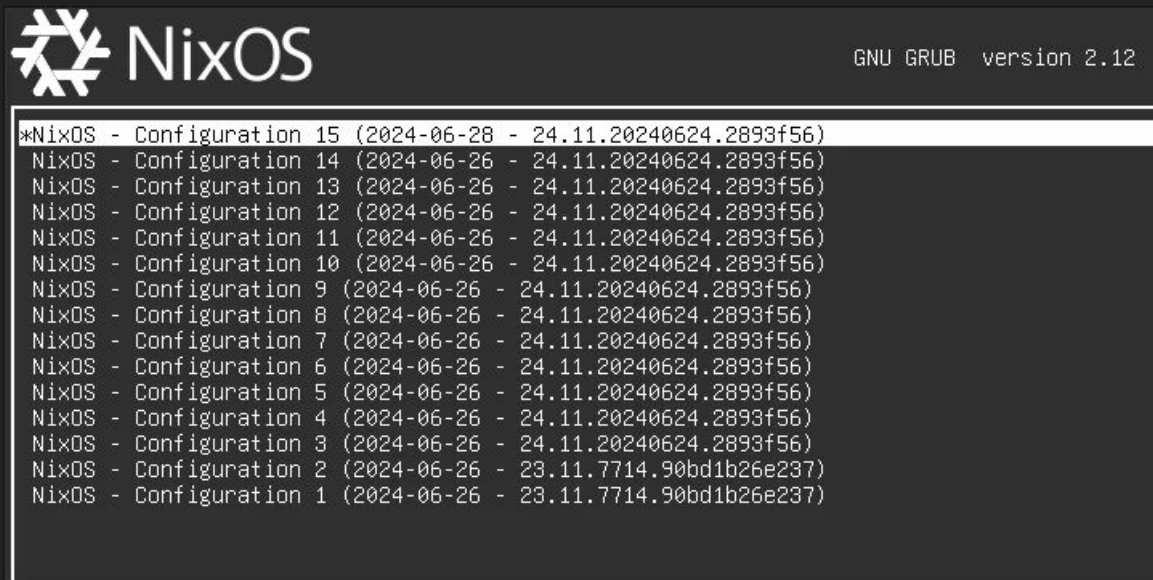
```
{
  services.nginx.enable = true;

  users.users.willie = {
    isNormalUser = true;
    extraGroups = [ "wheel" "networkmanager" ];
  };

  environment.systemPackages = with pkgs; [
    git
    vim
    firefox
  ];
}
```

Reliable

grub 可選 generation



```
NixOS GNU GRUB version 2.12
*NixOS - Configuration 15 (2024-06-28 - 24.11.20240624.2893f56)
NixOS - Configuration 14 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 13 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 12 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 11 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 10 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 9 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 8 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 7 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 6 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 5 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 4 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 3 (2024-06-26 - 24.11.20240624.2893f56)
NixOS - Configuration 2 (2024-06-26 - 23.11.7714.90bd1b26e237)
NixOS - Configuration 1 (2024-06-26 - 23.11.7714.90bd1b26e237)
```

Reliable

套件安裝位置

- 其他 Linux
 - /usr/bin
 - /usr/lib
 - /etc
- NixOS
 - /nix/store/abc123-openssl-3.0.13
 - /nix/store/def456-python-3.11.8
 - /nix/store/xyz789-nginx-1.24.0

NixOS 優缺點

優點

- 擁有最多套件
 - <https://repology.org/repositories/statistics/newest>
- 可快速回滾套件
- 環境可控、乾淨
- 系統可重現

缺點

- 幾乎無法運行非 Nix 的 Binary
- 與一般 Linux 差異很大

Nix 基礎語法

- <https://nix.dev/tutorials/nix-language>
 - functional programming

Nix 基礎語法

List

```
[  
  git  
  jq  
  python3  
]
```

用空白、換行分隔

不用逗號

Nix 基礎語法

變數、String

```
let
  name = "Nix";
in
  "Hello ${name}"
```

let ... in ... 定義變數

`${var_name}` 字串內插入變數

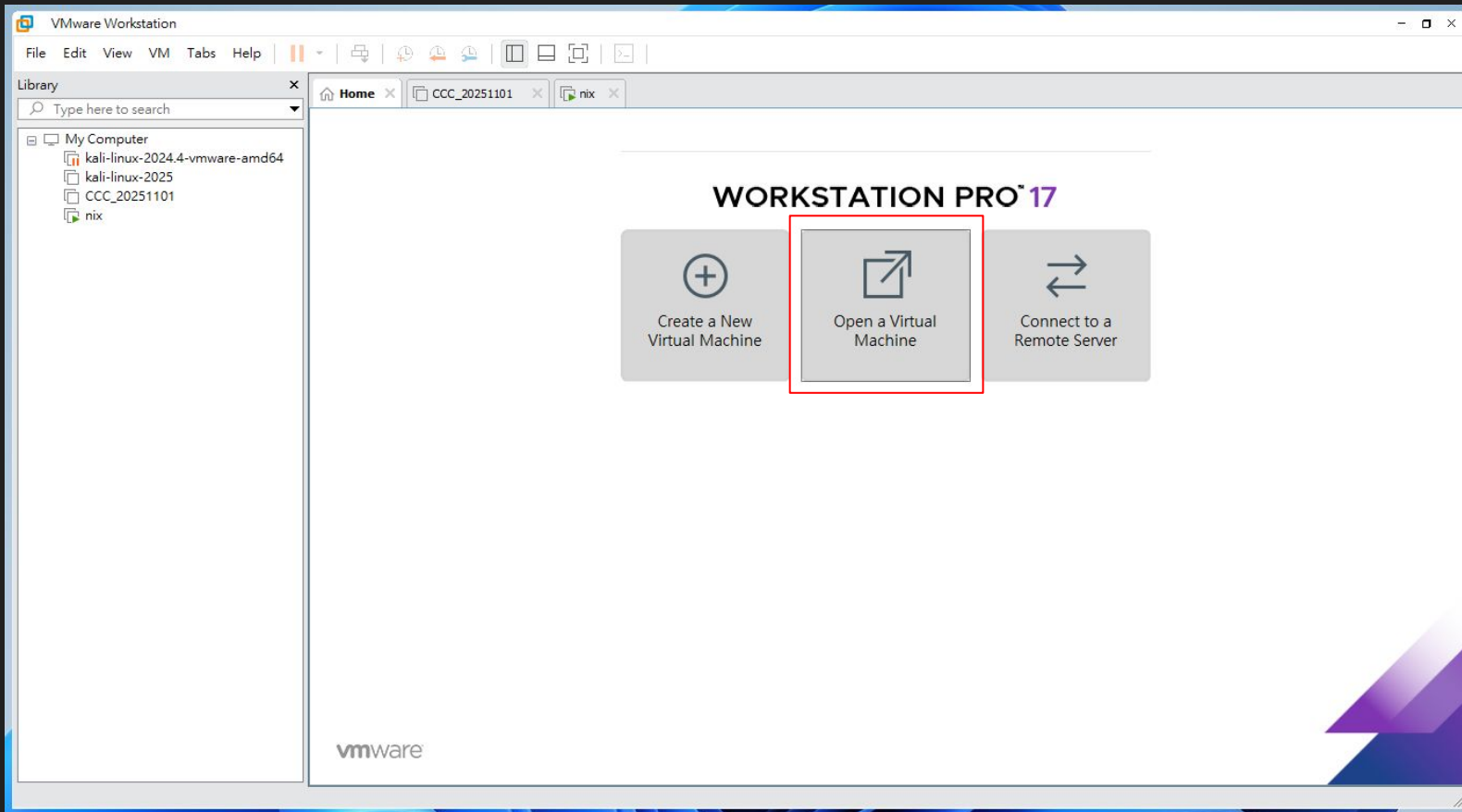
Nix 基礎語法

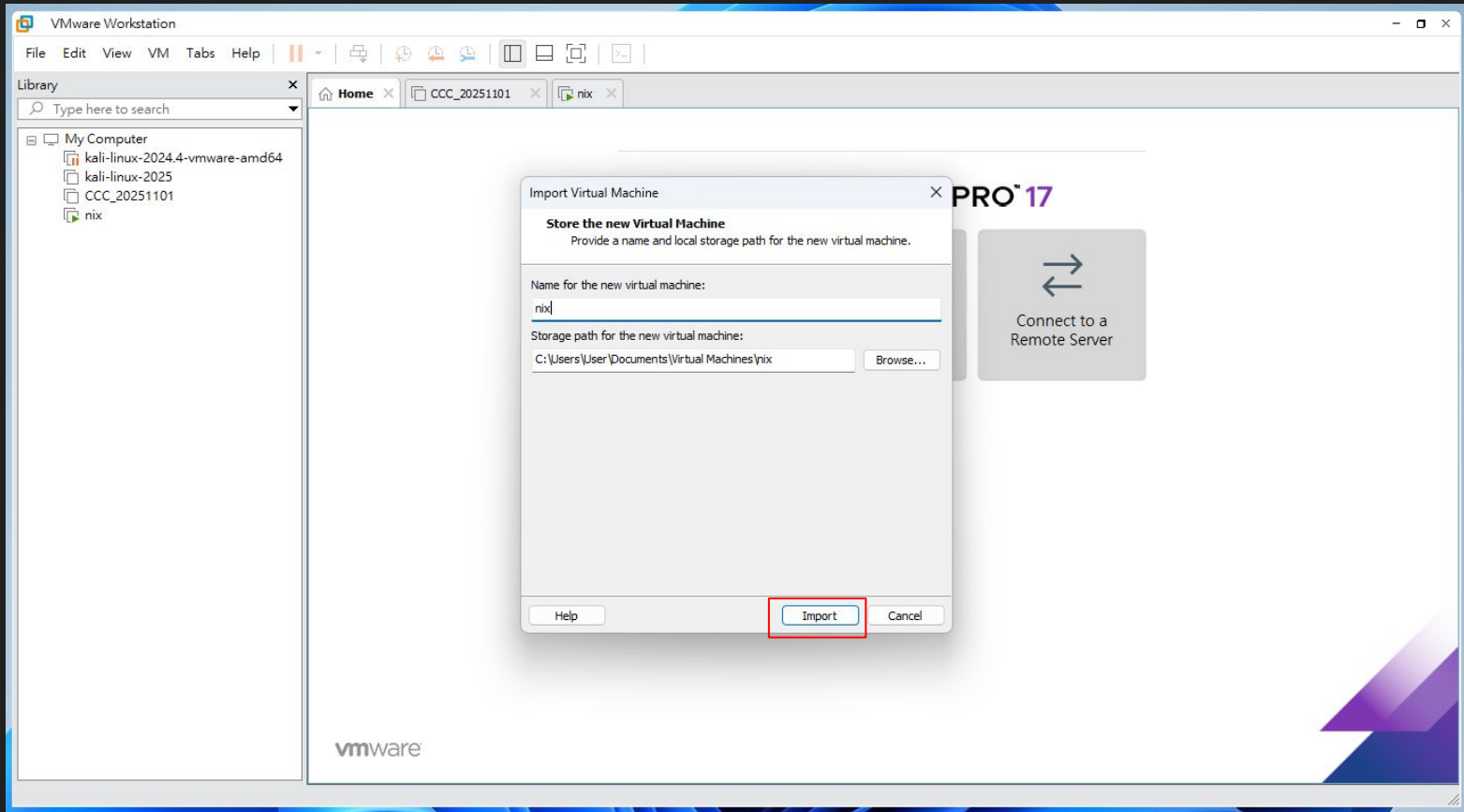
with ...; []

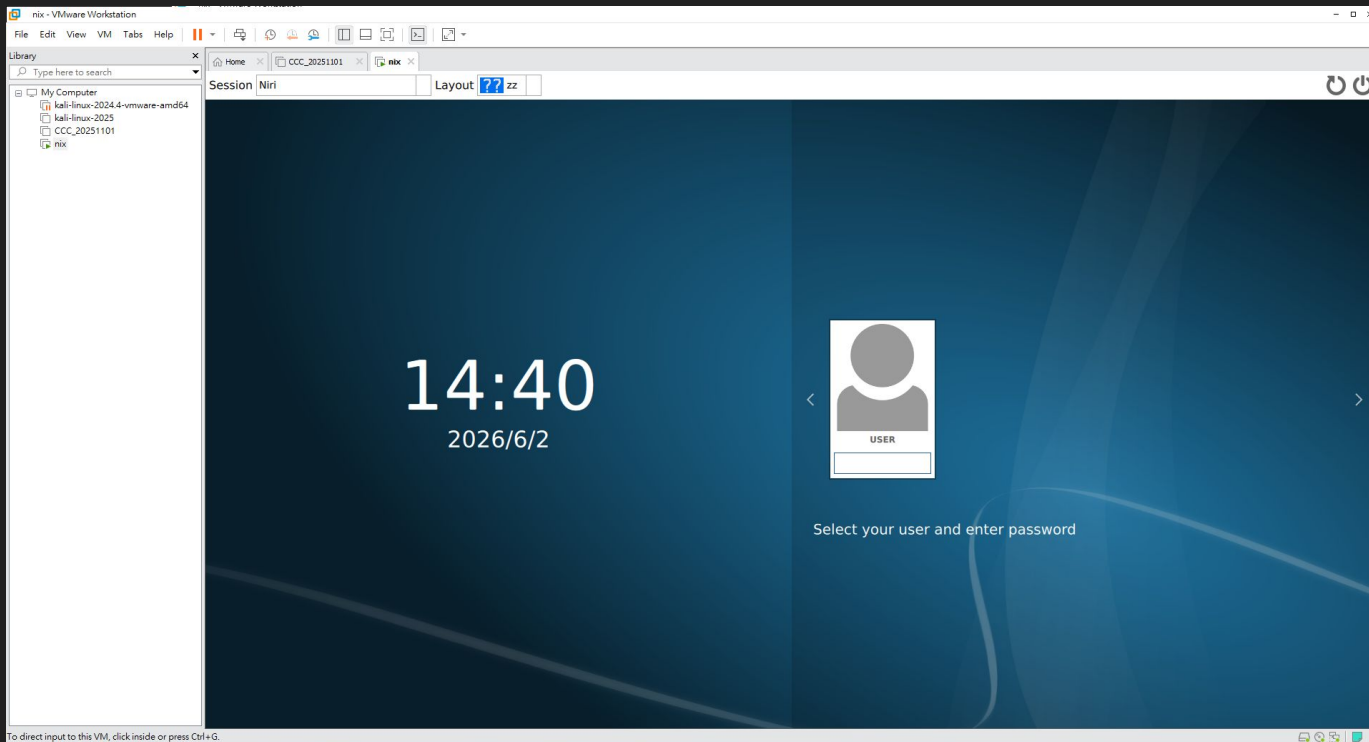
```
packages = with pkgs; [  
  git  
  jq  
  python3  
];
```

等同於

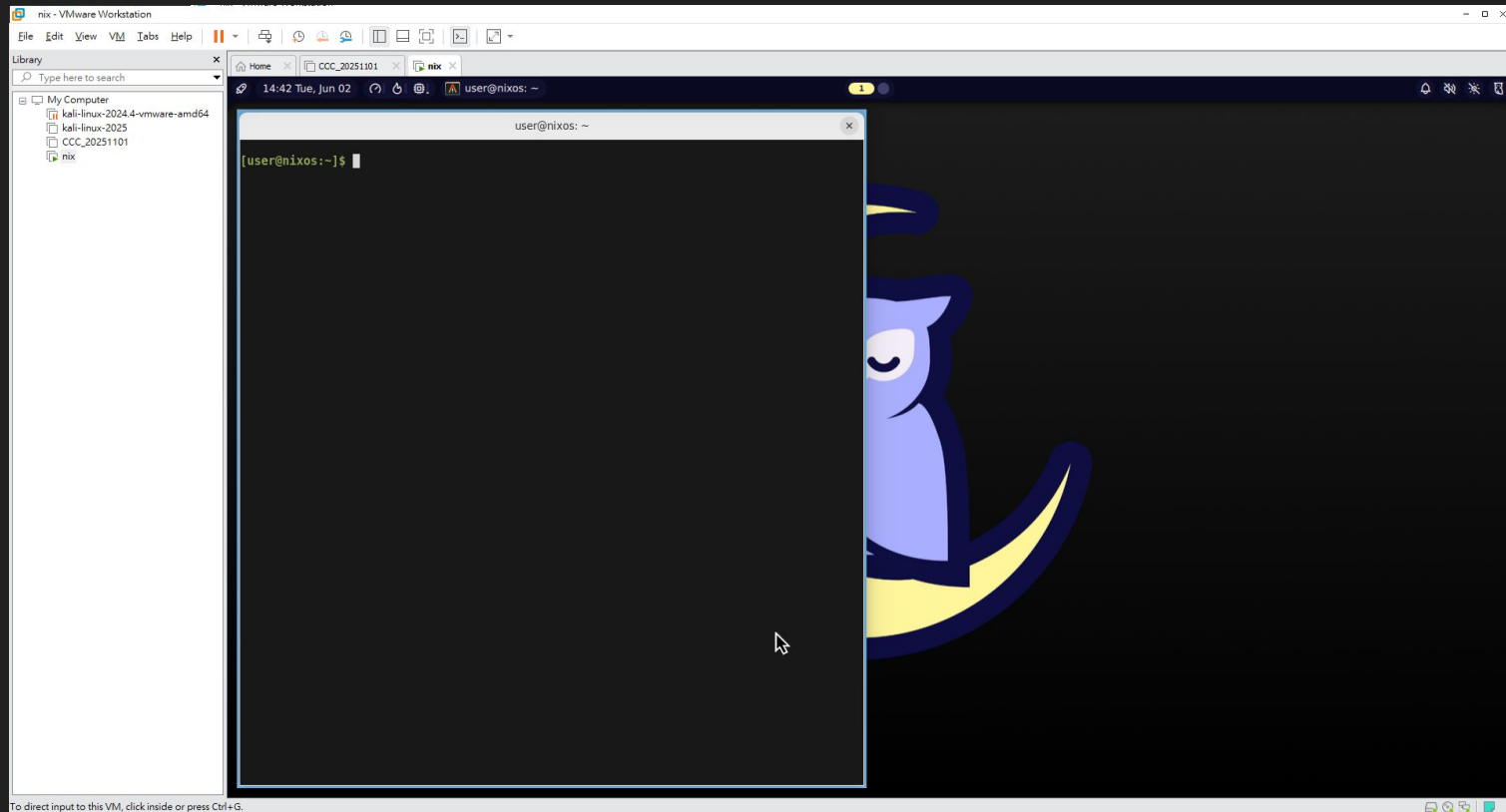
```
packages = [ pkgsg.git pkgs.jq pkgs.python3 ];
```







輸入密碼: 1234



To direct input to this VM, click inside or press Ctrl+G.

win + t: 開啟 terminal
win + f: 全螢幕
win + 滑鼠左右鍵: 拖曳、縮放視窗

左上角圖標: 開啟應用程式
右上角圖標: 開始選單? idk

```
sudo vi /etc/nixos/configuration.nix
```

定義整個系統組成

sudo vi /etc/nixos/configuration.nix

<https://search.nixos.org/packages>

安裝、解除安裝系統套件

```
environment.systemPackages = with pkgs; [  
    vim  
    git  
    alacritty  
    mako  
    mesa-demos  
    vulkan-tools  
    xwayland-satellite  
    noctalia-shell  
    thunar  
    vscode  
    #    firefox  
];
```

nixos-rebuild

- `sudo nixos-rebuild switch`
 - rebuild 目前 configuration 並直接切換至新 generation
- `sudo nixos-rebuild boot`
 - rebuild 目前 configuration, 在下次開機時才切換至新 generation
- `sudo nixos-rebuild list-generations`
 - 列出可用的 generation

開啟 ssh 服務、設定防火牆

```
services.openssh = {  
  enable = true;  
  ports = [ 22 ];  
  settings = {  
    PasswordAuthentication = true;  
    AllowUsers = null;  
    UseDns = true;  
    X11Forwarding = false;  
    PermitRootLogin = "prohibit-password";  
  };  
};
```

```
networking.firewall.allowedTCPPorts = [ 22 ];
```

flake

```
nix.settings.experimental-features = [ "flakes" ];
```

- 放置於 flake.nix
- 定義所有套件的出處
- 於 flake.lock 鎖定 repo 版本

```
{  
  description = "A very basic flake";  
  
  inputs = {  
    nixpkgs.url = "github:nixos/nixpkgs/nixos-26.05";  
  };  
  
  outputs = { self, nixpkgs }: {  
    nixosConfigurations.nixos = nixpkgs.lib.nixosSystem {  
      system = "x86_64-linux";  
  
      modules = [  
        ./configuration.nix  
      ];  
    };  
  };  
}
```

nix shell

- 舊版
 - `nix-shell -p cowsay`
- 新版
 - `nix shell nixpkgs#cowsay`

which cowsay

```
/nix/store/0r80s9lrgi64l53wrid79yymfyhqmqg8-cowsay-3.8.4/bin/cowsay
```

direnv

/etc/nixos/configuration.nix

```
programs.direnv.enable = true;
```

sudo nixos-rebuild switch

shell.nix

```
{ pkgs ? import <nixpkgs> {} }:
```

```
pkgs.mkShell {  
  packages = with pkgs; [  
    python3  
    hello  
  ];  
}
```

echo "use nix" >> .envrc
direnv allow

hello
python

home-manager

管理使用者層級設定、套件、dotfiles 與應用程式配置

Linux Ricing

What is Linux Ricing?

- Linux ricing 是指透過客製化桌面環境、視窗管理器、主題、圖示、字型與設定檔，把 Linux 系統調整成更美觀、更符合個人 workflow 與審美的過程
 - (By ChatGPT)
- <https://www.reddit.com/r/unixporn/>
- <https://www.reddit.com/r/niri/s/AYtG5sS7vs>

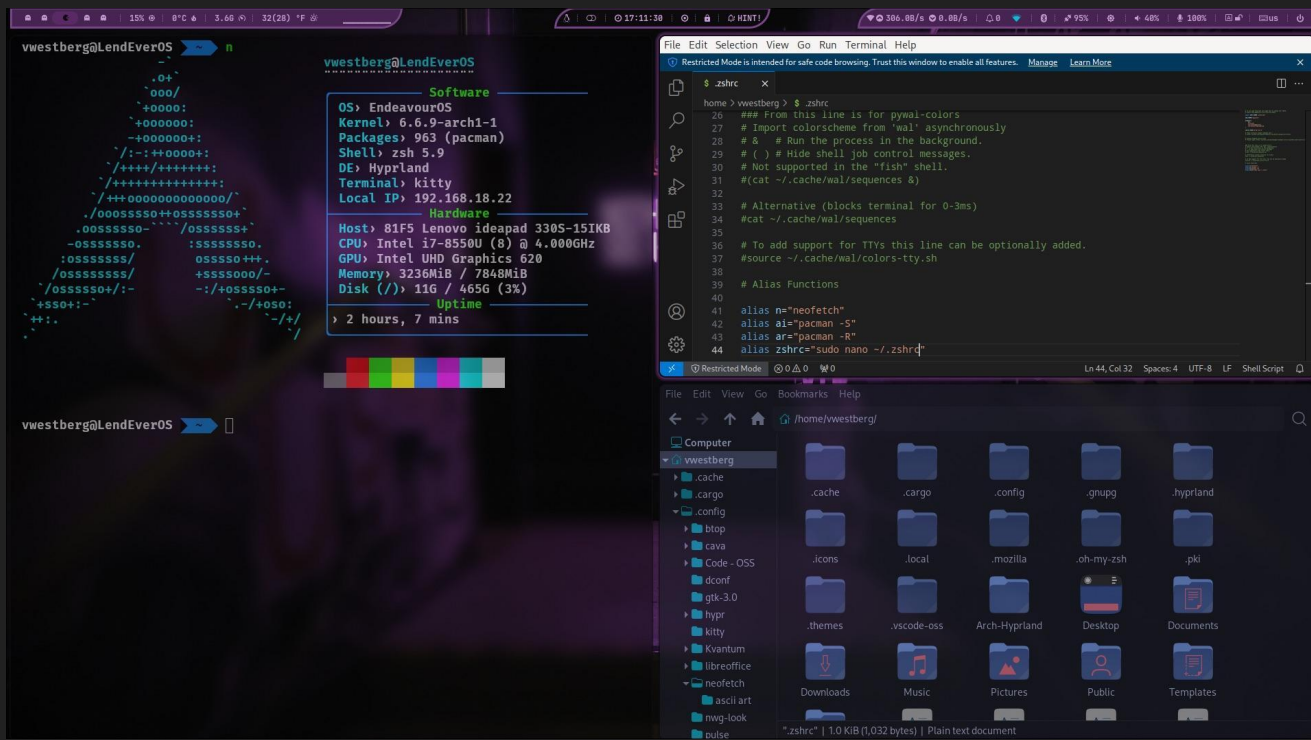
名詞解釋

- DE (Desktop Environment): 桌面環境
 - KDE Plasma: <https://kde.org/zh-tw/plasma-desktop/>
 - GNOME: <https://www.gnome.org/>
- WM (Window Manager): 視窗管理器
 - niri: <https://github.com/niri-wm/niri> (本次使用)
 - Hyprland: <https://hypr.land/>

名詞解釋

- Tiling Window: 平鋪視窗
 - niri: <https://github.com/niri-wm/niri> (本次使用)
 - Hyprland: <https://hypr.land/>
- Floating/Stacking Window: 堆疊視窗
 - Windows、macOS
 - KDE Plasma: KWin: <https://userbase.kde.org/KWin/>
 - GNOME: Mutter: <https://mutter.gnome.org/>

名詞解釋

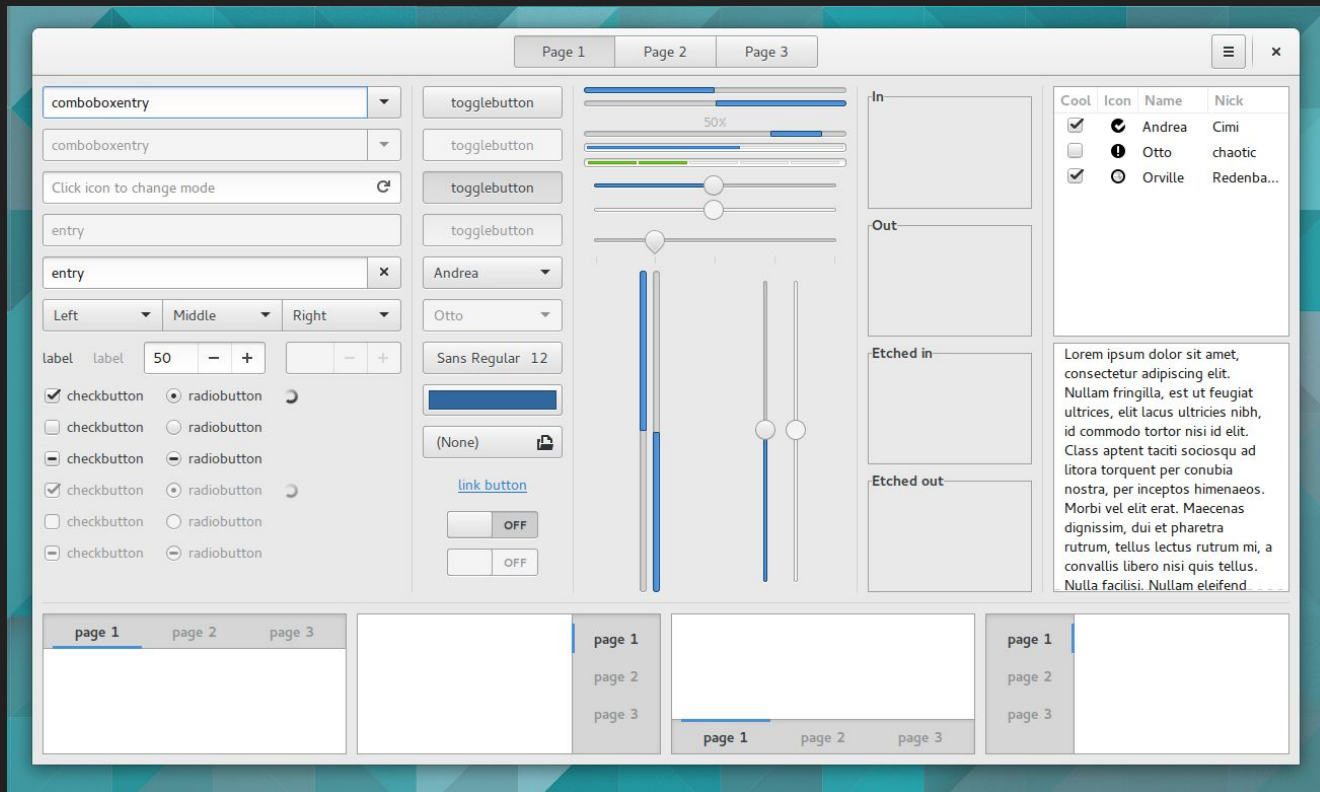


Tiling Window

GUI 框架

GTK: GNOME

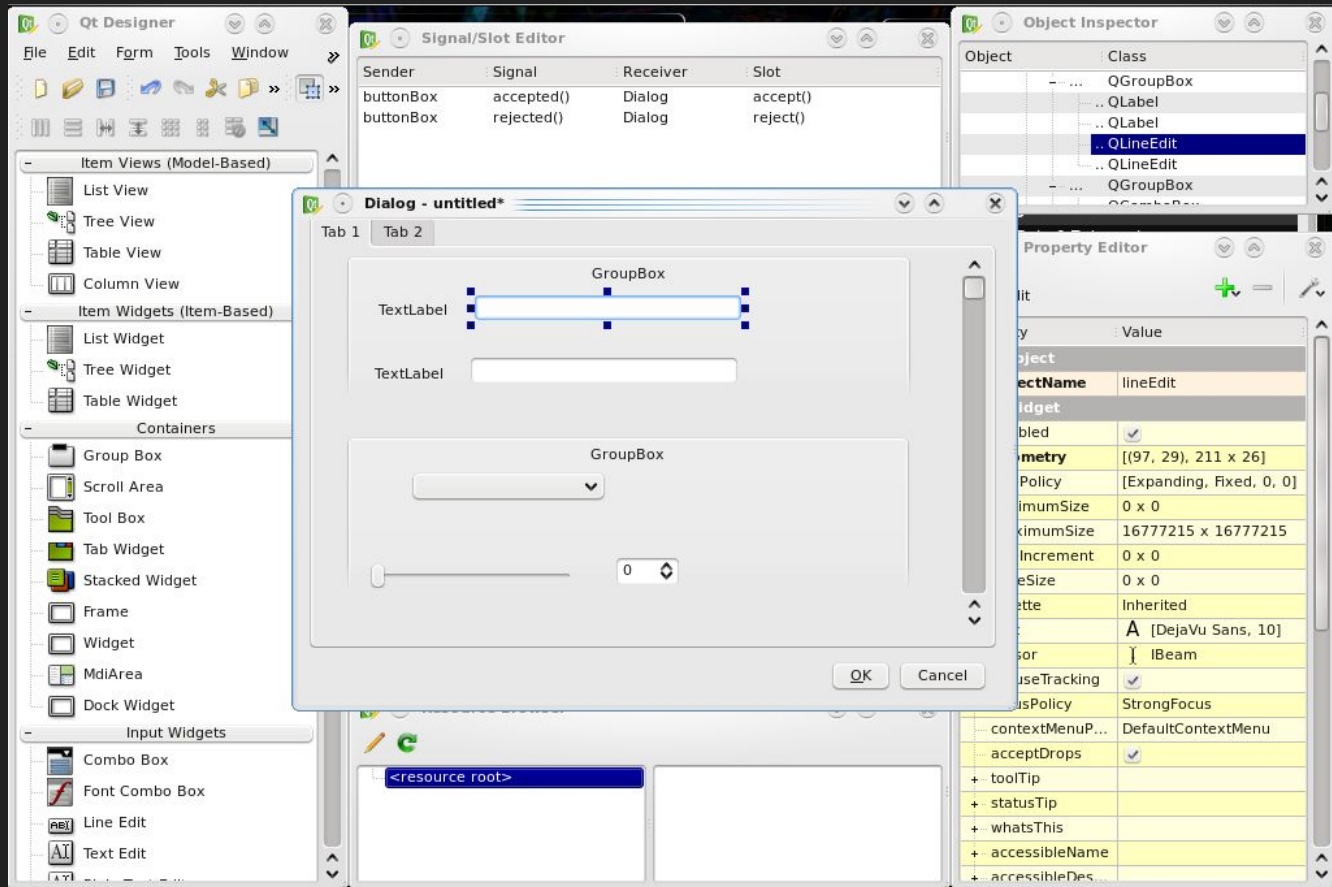
<https://www.gtk.org/>



GUI 框架

QT: KDE Plasma

<https://qt-project.org/>



名詞解釋

- dotfiles: 使用者 home 資料夾下 . 開頭的各個檔案 .config .local/share
- shell: 這裡是指覆蓋在視窗管理器/compositor 之上的桌面介面層
 - Quickshell: <https://quickshell.org/>
 - Noctalia Shell: <https://noctalia.dev/> (本次使用)
 - DankMaterialShell: <https://danklinux.com/>

Start Ricing !!!

- <https://github.com/catppuccin>
- <https://www.reddit.com/r/unixporn>
- <https://wallhaven.cc>
- <https://docs.noctalia.dev/v4/>

<https://devvillie.me/blog/hackersir-nix/>