

Guía Completa de Kitty Terminal

Edison Achalma

Escuela Profesional de Economía, Universidad Nacional de San Cristóbal de Huamanga

Resumen

Kitty es un emulador de terminal moderno, rápido y altamente personalizable que aprovecha la aceleración por GPU para ofrecer un rendimiento excepcional. Esta guía completa aborda desde la instalación en diferentes sistemas operativos hasta la configuración avanzada y optimización profunda de Kitty. Se exploran en detalle; arquitectura y principios de diseño, jerarquía de ventanas/pestañas, configuración de kitty.conf, manejo de fuentes con ligaduras, gestión avanzada de colores y temas, layouts nativos (tall, fat, grid, splits, stack), kittens (extensiones), control remoto, integración con shell, marcas, sesiones, optimización de rendimiento y solución de problemas frecuentes. Dirigida tanto a usuarios que buscan reemplazar terminales tradicionales por una opción más veloz y moderna, como a usuarios avanzados que desean exprimir al máximo las capacidades de personalización y automatización que ofrece Kitty en entornos de desarrollo y administración de sistemas.

Palabras Claves: Kitty Terminal, Linux, Línea de comandos

Tabla de contenidos

Introduction	6
1 Instalación	6
1.1 Requisitos del Sistema	6
1.2 Linux	7
1.3 macOS	7
1.4 Windows (WSL)	7
1.5 Verificar Instalación	7
1.6 Instalación de Dependencias Opcionales	7
2 Filosofía y Diseño	8
2.1 Principios de Diseño	8

Edison Achalma  <https://orcid.org/0000-0001-6996-3364>

El autor no tiene conflictos de interés que revelar. Los roles de autor se clasificaron utilizando la taxonomía de roles de colaborador (CRediT; <https://credit.niso.org/>) de la siguiente manera: Edison Achalma: conceptualización, redacción

La correspondencia relativa a este artículo debe dirigirse a Edison Achalma, Email: elmer.achalma.09@unsch.edu.pe

2.2	Arquitectura	8
3	Conceptos Básicos	8
3.1	Jerarquía: OS Windows > Tabs > Windows	8
3.2	Primer Uso	8
4	Configuración (kitty.conf)	9
4.1	Ubicación del Archivo	9
4.2	Generar Configuración Por Defecto	9
4.3	Estructura de Configuración	9
4.4	Configuración Básica Recomendada	9
4.5	Configuración Avanzada	13
5	Atajos de Teclado	14
5.1	Modificador Principal	14
5.2	Scrolling	14
5.3	Tabs	14
5.4	Windows (Ventanas dentro de Tabs)	15
5.5	Navegación Direccional	15
5.6	Layouts	15
5.7	Clipboard	15
5.8	Fuentes	16
5.9	Miscelánea	16
5.10	Atajos en macOS	16
6	Ventanas y Pestañas	16
6.1	Concepto de Multiplexación	16
6.2	Crear Ventanas y Tabs	17
6.3	Navegación	17
6.4	Mover Windows y Tabs	18
6.5	Detach (Separar)	18
6.6	Títulos Personalizados	19
7	Layouts (Diseños)	19
7.1	Layouts Disponibles	19
7.2	Stack Layout	19
7.3	Tall Layout	19
7.4	Fat Layout	20
7.5	Grid Layout	20
7.6	Splits Layout	21
7.7	Horizontal y Vertical Layouts	22
7.8	Cambiar Entre Layouts	22
7.9	Redimensionar Windows	22
8	Fuentes	23
8.1	Configuración Básica	23
8.2	Fuentes Recomendadas	23
8.3	Fuentes con Powerline	23
8.4	Symbol Maps	23

8.5	Ligaduras	24
8.6	Font Features	24
8.7	Modificar Fuentes	24
8.8	Kitten para Elegir Fuentes	25
9	Colores y Temas	25
9.1	Estructura de Colores	25
9.2	Temas Populares	26
9.3	Kitten Themes	27
9.4	Opacidad y Blur	28
9.5	Background Image	28
9.6	Dim Text	28
10	Cursor y Scrollback	28
10.1	Cursor	28
10.2	Scrollback	29
10.3	Scrollbar	30
11	Mouse	30
11.1	Comportamiento Básico	30
11.2	URLs	31
11.3	Selección	31
11.4	Pointer Shape	32
11.5	Mouse Actions	32
12	Kittens (Extensiones)	32
12.1	¿Qué son los Kittens?	32
12.2	Kittens Built-in	33
12.3	1. icat - Imágenes en Terminal	33
12.4	2. diff - Diff con Imágenes	33
12.5	3. unicode_input - Input Unicode	33
12.6	4. themes - Cambiar Temas	33
12.7	5. choose-fonts - Elegir Fuentes	34
12.8	6. hints - Selección Rápida	34
12.9	7. ssh - SSH Mejorado	35
12.10	8. transfer - Transferencia de Archivos	35
12.11	9. clipboard - Clipboard en SSH	36
12.12	10. broadcast - Broadcast Input	36
12.13	11. notify - Notificaciones	36
12.14	12. panel - Panel Desktop	36
12.15	13. @ (Remote Control)	36
12.16	Crear Kittens Personalizados	37
13	Control Remoto	37
13.1	Habilitar Remote Control	37
13.2	Uso Básico	37
13.3	Comandos Principales	38
13.4	Matching	39
13.5	Remote Control desde Mapeos	39

13.6 Remote Control via SSH	39
14 Sesiones	40
14.1 ¿Qué son las Sesiones?	40
14.2 Crear Sesión	40
14.3 Usar Sesión	40
14.4 Comandos de Sesión	41
14.5 Ejemplo Completo	42
14.6 Guardar Sesión Actual	42
14.7 Switching Sessions	42
15 Integración con Shell	43
15.1 Shell Integration	43
15.2 Habilitar	43
15.3 Configuración Manual	43
15.4 Características con Shell Integration	44
16 Marcas (Marks)	44
16.1 ¿Qué son las Marcas?	44
16.2 Tipos de Marcas	44
16.3 Configurar Colores	44
16.4 Crear Marcas	45
16.5 Usar Marcas	45
16.6 Ejemplo Completo	46
17 Características Avanzadas	46
17.1 Multiple Copy/Paste Buffers	46
17.2 Keyboard Modes	46
17.3 Per-Application Mappings	47
17.4 Environment Variables	47
17.5 Watchers	47
17.6 Filtros de Notificación	48
17.7 Allow Cloning	48
17.8 Notificaciones de Comando	48
18 Optimización de Rendimiento	48
18.1 Repaint Delay	48
18.2 Input Delay	48
18.3 Sync to Monitor	48
18.4 Scrollback	49
18.5 Text Composition	49
18.6 OpenGL	49
19 Trucos y Tips	49
19.1 Cambiar Directorio del Shell al Salir	49
19.2 Previsualizar Imágenes	49
19.3 Integración con Neovim	49
19.4 SSH con Túneles	50
19.5 Diff de Directorios	50

19.6 Broadcast Input	50
19.7 Quick Terminal (Quake-style)	50
19.8 8. URLs Personalizadas	50
19.9 Layouts Personalizados	50
19.10Themevar para Colores Dinámicos	51
20 Solución de Problemas	51
20.1 Problemas Comunes	51
20.2 Ligaduras No Funcionan	51
20.3 Colores Incorrectos	51
20.4 Imágenes No se Muestran	52
20.5 Performance Lento	52
20.6 Remote Control No Funciona	52
20.7 Fuentes Borrosas	53
20.8 Clipboard No Funciona	53
20.9 Debugging	53
21 Referencia Completa	54
21.1 Archivos de Configuración	54
21.2 Variables de Entorno	54
21.3 Códigos de Escape Útiles	54
21.4 API de Remote Control	54
21.5 Comparison Chart	55
22 Conclusión	55
22.1 Roadmap de Aprendizaje	55
22.2 Recursos	56
22.3 Tips Finales	56
23 Publicaciones Similares	56

Guía Completa de Kitty Terminal

¿Qué es Kitty?

Kitty es un emulador de terminal moderno, rápido y rico en características, diseñado desde cero para aprovechar las GPUs modernas. Creado por Kovid Goyal, es conocido por:

Características Principales:

- **Renderizado GPU acelerado:** Extremadamente rápido
- **Soporte Unicode completo:** Incluyendo emojis y ligaduras
- **True color:** 24-bit RGB
- **Ligaduras de fuentes:** Soporte nativo
- **Imágenes en terminal:** Protocolo propio de alta calidad
- **Multiplexación:** Tabs y splits nativos
- **Extensible:** Sistema de kittens (plugins)
- **Control remoto:** API completa
- **Configuración simple:** Archivo único y legible

Por Qué Usar Kitty

Ventajas sobre otros terminales:

Característica	Kitty	Alacritty	iTerm2	GNOME Terminal
Renderizado por GPU	Sí	Sí	Sí	No
Pestañas nativas	Sí	No	Sí	Sí
División de paneles	Sí	No	Sí	No
Soporte para imágenes	Sí	No	Sí	No
Ligaduras (ligatures)	Sí	Sí	Sí	Sí
Sistema de extensiones	Sí (kittens)	No	Sí	No
Control remoto (SSH)	Sí	No	Sí	No
Multiplataforma	Sí (Linux/macOS/Windows)	Sí (Linux/macOS/Windows)	No (solo macOS)	No (solo Linux)

1 Instalación

1.1 Requisitos del Sistema

Mínimos:

- OpenGL 3.3+
- Linux, macOS, o Windows (WSL)
- Python 3.6+

Recomendados:

- GPU moderna con drivers actualizados
- Monitor de alta resolución (4K/Retina)
- Fuente con ligaduras (FiraCode, JetBrains Mono)

1.2 Linux

Ubuntu/Debian:

```
sudo apt install kitty
```

Arch Linux:

```
sudo pacman -S kitty
```

Fedora:

```
sudo dnf install kitty
```

Desde binario oficial (recomendado):

```
curl -L https://sw.kovidgoyal.net/kitty/installer.sh | sh /dev/stdin
```

1.3 macOS

Homebrew:

```
brew install --cask kitty
```

Desde DMG:

Descargar desde: <https://github.com/kovidgoyal/kitty/releases>

1.4 Windows (WSL)

```
curl -L https://sw.kovidgoyal.net/kitty/installer.sh | sh /dev/stdin
```

1.5 Verificar Instalación

```
kitty --version
```

1.6 Instalación de Dependencias Opcionales

```
# Herramientas útiles para kittens
sudo apt install imagemagick      # Procesamiento de imágenes
sudo apt install bat              # Mejor syntax highlighting
sudo apt install fzf              # Fuzzy finder
sudo apt install ripgrep          # Búsqueda rápida
```

2 Filosofía y Diseño

2.1 Principios de Diseño

1. **Velocidad:** GPU rendering para máximo rendimiento
2. **Simplicidad:** Configuración en un solo archivo
3. **Extensibilidad:** Kittens para añadir funcionalidad
4. **Estándares:** Soporte completo de especificaciones modernas
5. **Teclado primero:** Control total desde teclado

2.2 Arquitectura

Componentes principales:

- **Core (C):** Renderizado GPU, performance crítico
- **UI (Python):** Interfaz, configuración, extensibilidad
- **Kittens (Python/Go):** Programas auxiliares

Stack tecnológico:

- OpenGL para renderizado
- HarfBuzz para text shaping
- FontConfig para gestión de fuentes
- libunistring para Unicode

3 Conceptos Básicos

3.1 Jerarquía: OS Windows > Tabs > Windows

OS Window (ventana del sistema)

Tab 1 Tab 2 Tab 3

Window 1 Window 2

OS Window: Ventana del sistema operativo **Tab:** Pestaña dentro de un OS Window
Window: Panel dentro de un Tab (como splits en tmux)

3.2 Primer Uso

Al abrir Kitty por primera vez:

```
# Abrir Kitty
kitty

# Ver ayuda
Ctrl+Shift+F1

# Editar configuración
Ctrl+Shift+F2

# Recargar configuración
Ctrl+Shift+F5
```

4 Configuración (kitty.conf)

4.1 Ubicación del Archivo

Linux/macOS: ~/.config/kitty/kitty.conf
Windows: %LOCALAPPDATA%\kitty\kitty.conf

4.2 Generar Configuración Por Defecto

```
# Generar con comentarios
kitty +runpy 'from kitty.config import *; print(commented_out_default_config())'

# 0 abrir para editar
Ctrl+Shift+F2
```

4.3 Estructura de Configuración

Secciones principales:

1. Fuentes
2. Cursor
3. Scrollback
4. Mouse
5. Terminal bell
6. Window layout
7. Tab bar
8. Color scheme
9. Advanced
10. OS tweaks
11. Keyboard shortcuts

4.4 Configuración Básica Recomendada

```
# =====
# CONFIGURACIÓN BÁSICA DE KITTY
# =====
```

```
# Fuentes
font_family      JetBrains Mono
bold_font        auto
italic_font       auto
bold_italic_font auto
font_size        12.0

# Cursor
cursor_shape block
cursor_blink_interval 0
cursor_stop_blinking_after 15.0

# Scrollback
scrollback_lines 10000
scrollback_pager less --chop-long-lines --RAW-CONTROL-CHARS +INPUT_LINE_NUMBER

# Mouse
mouse_hide_wait 3.0
url_color #0087bd
url_style curly
open_url_with default
detect_urls yes
copy_on_select no

# Ventanas
remember_window_size yes
initial_window_width 640
initial_window_height 400
window_border_width 0.5pt
window_margin_width 0
window_padding_width 0
active_border_color #00ff00
inactive_border_color #cccccc

# Pestañas
tab_bar_edge bottom
tab_bar_style fade
tab_bar_min_tabs 2
tab_title_template "{index}: {title}"

# Colores (Dracula theme)
foreground #f8f8f2
background #282a36
selection_foreground #ffffff
selection_background #44475a

# Negro
```

```
color0 #21222c
color8 #6272a4

# Rojo
color1 #ff5555
color9 #ff6e6e

# Verde
color2 #50fa7b
color10 #69ff94

# Amarillo
color3 #f1fa8c
color11 #ffffa5

# Azul
color4 #bd93f9
color12 #d6acff

# Magenta
color5 #ff79c6
color13 #ff92df

# Cyan
color6 #8be9fd
color14 #a4ffff

# Blanco
color7 #f8f8f2
color15 #ffffff

# Rendimiento
repaint_delay 10
input_delay 3
sync_to_monitor yes

# Bell
enable_audio_bell no
visual_bell_duration 0.0
window_alert_on_bell yes
bell_on_tab " "

# Advanced
shell .
editor .
allow_remote_control yes
listen_on unix:/tmp/kitty
```

```
clipboard_control write-clipboard write-primary read-clipboard-ask read-primary-ask

# OS Específico (macOS)
macos_titlebar_color system
macos_option_as_alt no
macos_hide_from_tasks no
macos_quit_when_last_window_closed no

# Layouts habilitados
enabled_layouts tall:bias=50;full_size=1,stack,fat:bias=50;full_size=1,grid,splits

# =====
# MAPEOS DE TECLADO
# =====

# Modificador principal
kitty_mod ctrl+shift

# Clipboard
map kitty_mod+c copy_to_clipboard
map kitty_mod+v paste_from_clipboard
map kitty_mod+s paste_from_selection

# Scrolling
map kitty_mod+up scroll_line_up
map kitty_mod+down scroll_line_down
map kitty_mod+page_up scroll_page_up
map kitty_mod+page_down scroll_page_down
map kitty_mod+home scroll_home
map kitty_mod+end scroll_end
map kitty_mod+h show_scrollback
map kitty_mod+g show_last_command_output

# Ventanas
map kitty_mod+enter new_window
map kitty_mod+n new_os_window
map kitty_mod+w close_window
map kitty_mod+] next_window
map kitty_mod+[ previous_window
map kitty_mod+f move_window_forward
map kitty_mod+b move_window_backward
map kitty_mod+` move_window_to_top
map kitty_mod+r start_resizing_window
map kitty_mod+1 first_window
map kitty_mod+2 second_window
map kitty_mod+3 third_window
map kitty_mod+4 fourth_window
```

```

map kitty_mod+5 fifth_window

# Tabs
map kitty_mod+t new_tab
map kitty_mod+q close_tab
map kitty_mod+right next_tab
map kitty_mod+left previous_tab
map kitty_mod+. move_tab_forward
map kitty_mod+, move_tab_backward
map kitty_mod+alt+t set_tab_title

# Layouts
map kitty_mod+l next_layout

# Fuentes
map kitty_mod+equal change_font_size all +2.0
map kitty_mod+minus change_font_size all -2.0
map kitty_mod+backspace change_font_size all 0

# Misc
map kitty_mod+f1 show_kitty_doc overview
map kitty_mod+f2 edit_config_file
map kitty_mod+f5 load_config_file
map kitty_mod+f6 debug_config
map kitty_mod+f11 toggle_fullscreen
map kitty_mod+f10 toggle_maximized
map kitty_mod+u kitten unicode_input
map kitty_mod+e open_url_with_hints
map kitty_mod+escape kitty_shell window

# Opacidad de fondo
map kitty_mod+a>m set_background_opacity +0.1
map kitty_mod+a>l set_background_opacity -0.1
map kitty_mod+a>1 set_background_opacity 1
map kitty_mod+a>d set_background_opacity default

# Reset terminal
map kitty_mod+delete clear_terminal reset active

```

4.5 Configuración Avanzada

Fuentes con fallbacks:

```

font_family      JetBrains Mono
# Símbolos especiales
symbol_map U+E0A0-U+E0A3,U+E0C0-U+E0C7 PowerlineSymbols
# Emojis
symbol_map U+1F300-U+1F9FF Noto Color Emoji

```

Ligaduras:

```
# Habilitar ligaduras
disable_ligatures never
# 0 solo cuando el cursor no está sobre ellas
# disable_ligatures cursor
```

Modificar fuentes:

```
# Ajustar posición del underline
modify_font underline_position 2
modify_font underline_thickness 150%
# Ajustar tamaño de celda
modify_font cell_width 100%
modify_font cell_height 100%
```

5 Atajos de Teclado**5.1 Modificador Principal**

Por defecto kitty_mod = Ctrl+Shift
En macOS algunos también usan Cmd ()

5.2 Scrolling

Atajo	Acción
Ctrl+Shift+Up	Scroll línea arriba
Ctrl+Shift+Down	Scroll línea abajo
Ctrl+Shift+Page Up	Scroll página arriba
Ctrl+Shift+Page Down	Scroll página abajo
Ctrl+Shift+Home	Ir al inicio
Ctrl+Shift+End	Ir al final
Ctrl+Shift+Z	Prompt anterior (shell integration)
Ctrl+Shift+X	Prompt siguiente (shell integration)
Ctrl+Shift+H	Ver scrollbar en pager
Ctrl+Shift+G	Ver último comando (shell integration)
Ctrl+Shift+/,	Buscar en scrollbar

5.3 Tabs

Atajo	Acción
Ctrl+Shift+T	Nueva tab
Ctrl+Shift+Q	Cerrar tab
Ctrl+Shift+Right	Tab siguiente
Ctrl+Shift+Left	Tab anterior
Ctrl+Shift+.,	Mover tab adelante
Ctrl+Shift+,,	Mover tab atrás

Atajo	Acción
Ctrl+Shift+Alt+T	Renombrar tab
Ctrl+Alt+1...9	Ir a tab N

5.4 Windows (Ventanas dentro de Tabs)

Atajo	Acción
Ctrl+Shift+Enter	Nueva window
Ctrl+Shift+N	Nueva OS window
Ctrl+Shift+W	Cerrar window
Ctrl+Shift+]	Window siguiente
Ctrl+Shift+[	Window anterior
Ctrl+Shift+F	Mover window adelante
Ctrl+Shift+B	Mover window atrás
Ctrl+Shift+``' Mover window al top Ctrl+Shift+R Redimensionar window Ctrl+Shift+F7 Selección visual de window Ctrl+Shift+F8 Swap visual de window Ctrl+Shift+1...0'	Ir a window N

5.5 Navegación Direccional

Configurar manualmente:

```
# Navegar entre windows
map ctrl+left neighboring_window left
map ctrl+right neighboring_window right
map ctrl+up neighboring_window up
map ctrl+down neighboring_window down

# Mover windows
map shift+left move_window right
map shift+right move_window left
map shift+up move_window down
map shift+down move_window up
```

5.6 Layouts

Atajo	Acción
Ctrl+Shift+L	Siguiente layout

5.7 Clipboard

Atajo	Acción
Ctrl+Shift+C	Copiar
Ctrl+Shift+V	Pegar
Ctrl+Shift+S	Pegar de selección
Ctrl+Shift+O	Pasar selección a programa

5.8 Fuentes

Atajo	Acción
Ctrl+Shift+Equal	Aumentar tamaño
Ctrl+Shift+Minus	Disminuir tamaño
Ctrl+Shift+Backspace	Reset tamaño

5.9 Miscelánea

Atajo	Acción
Ctrl+Shift+F1	Mostrar ayuda
Ctrl+Shift+F2	Editar config
Ctrl+Shift+F5	Recargar config
Ctrl+Shift+F6	Debug config
Ctrl+Shift+F11	Toggle fullscreen
Ctrl+Shift+F10	Toggle maximized
Ctrl+Shift+U	Input Unicode
Ctrl+Shift+E	Open URL hints
Ctrl+Shift+Escape	Kitty shell
Ctrl+Shift+Delete	Reset terminal

5.10 Atajos en macOS

Reemplazan algunos con Cmd ():

Windows/Linux	macOS
Ctrl+Shift+T	Cmd+T
Ctrl+Shift+W	Cmd+W
Ctrl+Shift+C	Cmd+C
Ctrl+Shift+V	Cmd+V
Ctrl+Shift+N	Cmd+N
Ctrl+Shift+F2	Cmd+,
Ctrl+Shift+F11	Ctrl+Cmd+F

6 Ventanas y Pestañas

6.1 Concepto de Multiplexación

Kitty tiene multiplexación nativa (sin necesidad de tmux):

Jerarquía:

OS Window (kitty instance)

```
Tab 1
  Window 1
  Window 2
  Window 3
Tab 2
  Window 1
```

6.2 Crear Ventanas y Tabs

Nueva window:

```
# Atajo
Ctrl+Shift+Enter

# Comando
kitty @ launch --type=window

# Con directorio específico
map f1 launch --cwd=current
```

Nueva tab:

```
# Atajo
Ctrl+Shift+T

# Con programa
map f1 new_tab_with_cwd
```

Nueva OS window:

```
Ctrl+Shift+N
```

6.3 Navegación

Entre windows:

```
# Direccional
map ctrl+left neighboring_window left
map ctrl+right neighboring_window right
map ctrl+up neighboring_window up
map ctrl+down neighboring_window down

# Numérica
map ctrl+shift+1 first_window
map ctrl+shift+2 second_window
# ...
```

Entre tabs:

```
map ctrl+shift+right next_tab
map ctrl+shift+left previous_tab
map ctrl+alt+1 goto_tab 1
```

6.4 Mover Windows y Tabs

Mover windows:

```
map ctrl+shift+f move_window_forward
map ctrl+shift+b move_window_backward
map ctrl+shift+` move_window_to_top

# Direccional
map shift+up move_window up
map shift+down move_window down
map shift+left move_window left
map shift+right move_window right
```

Mover tabs:

```
map ctrl+shift+. move_tab_forward
map ctrl+shift+, move_tab_backward
```

6.5 Detach (Separar)

Detach window:

```
# A nueva OS window
map ctrl+f2 detach_window

# A nueva tab
map ctrl+f3 detach_window new-tab

# A tab anterior
map ctrl+f3 detach_window tab-prev

# Preguntar destino
map ctrl+f4 detach_window ask
```

Detach tab:

```
# A nueva OS window
map ctrl+f2 detach_tab

# Preguntar destino
map ctrl+f4 detach_tab ask
```

6.6 Títulos Personalizados

```
# Renombrar tab
map ctrl+shift+alt+t set_tab_title

# Renombrar window
map f1 set_window_title

# Template de título
tab_title_template "{index}: {title}"
```

7 Layouts (Diseños)

7.1 Layouts Disponibles

Kitty tiene 7 layouts built-in:

1. **Stack**: Una window a la vez (maximizada)
2. **Tall**: Una o más windows full-height a la izquierda
3. **Fat**: Una o más windows full-width arriba
4. **Grid**: Grid balanceado
5. **Splits**: Arreglo arbitrario (como tmux)
6. **Horizontal**: Todas las windows lado a lado
7. **Vertical**: Todas las windows una debajo de otra

7.2 Stack Layout

Window Actual
(maximizada)

Configuración:

```
enabled_layouts stack
```

7.3 Tall Layout

Full
Height

Configuración:

```
enabled_layouts tall:bias=50;full_size=1;mirrored=false

# Opciones:
# bias: 10-90 (split horizontal)
# full_size: número de windows full-height
# mirrored: true/false (full-height a la derecha)
```

Acciones:

```
# Aumentar/disminuir windows full-height
map ctrl+[ layout_action decrease_num_full_size_windows
map ctrl+] layout_action increase_num_full_size_windows

# Toggle mirror
map ctrl+/ layout_action mirror toggle

# Cambiar bias
map ctrl+. layout_action bias 50 62 70
```

7.4 Fat Layout

Full Width

Configuración:

```
enabled_layouts fat:bias=50;full_size=1;mirrored=false
```

Mismas acciones que Tall layout.

7.5 Grid Layout

Configuración:

```
enabled_layouts grid
```

7.6 Splits Layout

El más flexible - como tmux/vim splits.

Configuración:

```
enabled_layouts splits:split_axis=horizontal  
# split_axis: horizontal, vertical, auto
```

Crear splits:

```
# Split horizontal  
map f5 launch --location=hsplit  
  
# Split vertical  
map f6 launch --location=vsplit  
  
# Split auto (horizontal si ancho, vertical si alto)  
map f4 launch --location=split  
  
# Rotar split (horizontal <-> vertical)  
map f7 layout_action rotate  
  
# Mover window  
map shift+up move_window up  
map shift+down move_window down  
map shift+left move_window left  
map shift+right move_window right  
  
# Mover window al borde  
map ctrl+shift+up layout_action move_to_screen_edge top  
map ctrl+shift+down layout_action move_to_screen_edge bottom  
map ctrl+shift+left layout_action move_to_screen_edge left  
map ctrl+shift+right layout_action move_to_screen_edge right
```

```
# Bias del split
map ctrl+. layout_action bias 80
```

7.7 Horizontal y Vertical Layouts

Horizontal (lado a lado):

Vertical (uno debajo del otro):

Configuración:

```
enabled_layouts horizontal,vertical
```

7.8 Cambiar Entre Layouts

```
# Siguiente layout
map ctrl+shift+l next_layout

# Ir a layout específico
map ctrl+alt+t goto_layout tall
map ctrl+alt+s goto_layout stack

# Toggle layout
map ctrl+alt+z toggle_layout stack
```

7.9 Redimensionar Windows

Modo interactivo:

```
Ctrl+Shift+R
# Seguir instrucciones en pantalla
```

Atajos directos:

```
map ctrl+left resize_window narrower
map ctrl+right resize_window wider
map ctrl+up resize_window taller
map ctrl+down resize_window shorter 3
map ctrl+home resize_window reset
```

8 Fuentes

8.1 Configuración Básica

```
font_family      JetBrains Mono
bold_font        auto
italic_font       auto
bold_italic_font auto
font_size        12.0
```

8.2 Fuentes Recomendadas

Con ligaduras:

- JetBrains Mono
- Fira Code
- Cascadia Code
- Victor Mono
- Iosevka

Mono clásicas:

- IBM Plex Mono
- Source Code Pro
- Ubuntu Mono
- Inconsolata

8.3 Fuentes con Powerline

```
# Método 1: Usar font con powerline incluido
font_family JetBrains Mono Nerd Font

# Método 2: Symbol map
font_family JetBrains Mono
symbol_map U+E0A0-U+E0A3,U+E0C0-U+E0C7 PowerlineSymbols
```

8.4 Symbol Maps

Mapear rangos Unicode a fuentes específicas:

```
# Powerline
symbol_map U+E0A0-U+E0A3,U+E0C0-U+E0C7 PowerlineSymbols

# Emojis
symbol_map U+1F300-U+1F64F,U+1F680-U+1F6FF Noto Color Emoji

# Math
symbol_map U+2200-U+22FF STIX Two Math
```

8.5 Ligaduras

```
# Siempre habilitadas
disable_ligatures never

# Deshabilitadas cuando cursor está sobre ellas
disable_ligatures cursor

# Siempre deshabilitadas
disable_ligatures always
```

Controlar por window:

```
map alt+1 disable_ligatures_in active always
map alt+2 disable_ligatures_in all never
map alt+3 disable_ligatures_in tab cursor
```

8.6 Font Features

```
# Habilitar features específicas
# Ejemplo con Fira Code:
font_features FiraCode-Regular +zero +onum +ss01 +ss02

# Deshabilitar ligatures pero mantener calt
font_features FiraCode-Regular -liga +calt
```

8.7 Modificar Fuentes

```
# Ajustar underline
modify_font underline_position -2
modify_font underline_thickness 150%

# Ajustar strikethrough
modify_font strikethrough_position 2px

# Ajustar tamaño de celda
```

```
modify_font cell_width 80%
modify_font cell_height -2px

# Ajustar baseline
modify_font baseline 3
```

8.8 Kitten para Elegir Fuentes

```
# UI interactiva para elegir fuentes
kitten choose-fonts

# Previsualizar todas las fuentes
kitten choose-fonts --preview
```

9 Colores y Temas

9.1 Estructura de Colores

```
# Colores básicos
foreground #foreground_color
background #background_color

# Selección
selection_foreground #color
selection_background #color

# Cursor
cursor #color
cursor_text_color #color

# Tabla de colores (16 básicos)
# Negro
color0 #color
color8 #color

# Rojo
color1 #color
color9 #color

# Verde
color2 #color
color10 #color

# Amarillo
color3 #color
color11 #color
```

```
# Azul
color4 #color
color12 #color

# Magenta
color5 #color
color13 #color

# Cyan
color6 #color
color14 #color

# Blanco
color7 #color
color15 #color

# Colores extendidos (256 colores)
color16 #color
color17 #color
# ... hasta color255
```

9.2 Temas Populares

Dracula:

```
foreground #f8f8f2
background #282a36
selection_foreground #ffffff
selection_background #44475a

color0 #21222c
color8 #6272a4
color1 #ff5555
color9 #ff6e6e
color2 #50fa7b
color10 #69ff94
color3 #f1fa8c
color11 #ffffa5
color4 #bd93f9
color12 #d6acff
color5 #ff79c6
color13 #ff92df
color6 #8be9fd
color14 #a4ffff
color7 #f8f8f2
color15 #ffffff
```

Nord:

```
foreground #d8dee9
background #2e3440

color0 #3b4252
color8 #4c566a
color1 #bf616a
color9 #bf616a
color2 #a3be8c
color10 #a3be8c
color3 #ebcb8b
color11 #ebcb8b
color4 #81a1c1
color12 #81a1c1
color5 #b48ead
color13 #b48ead
color6 #88c0d0
color14 #8fbcbb
color7 #e5e9f0
color15 #eceff4
```

Gruvbox Dark:

```
foreground #ebdbb2
background #282828

color0 #282828
color8 #928374
color1 #cc241d
color9 #fb4934
color2 #98971a
color10 #b8bb26
color3 #d79921
color11 #fabd2f
color4 #458588
color12 #83a598
color5 #b16286
color13 #d3869b
color6 #689d6a
color14 #8ec07c
color7 #a89984
color15 #ebdbb2
```

9.3 Kitten Themes

```
# Browser de temas interactivo
kitten themes
```

```
# Listar temas disponibles
kitten themes --list

# Aplicar tema específico
kitten themes --reload-in=all Dracula
```

9.4 Opacidad y Blur

```
# Opacidad del background
background_opacity 0.9

# Background blur (macOS y KDE)
background_blur 20

# Dynamic opacity (permite cambiar con atajos)
dynamic_background_opacity yes

# Cambiar opacidad con atajos
map ctrl+shift+a>m set_background_opacity +0.1
map ctrl+shift+a>l set_background_opacity -0.1
map ctrl+shift+a>1 set_background_opacity 1
map ctrl+shift+a>d set_background_opacity default
```

9.5 Background Image

```
background_image /path/to/image.png
background_image_layout tiled
# Opciones: tiled, mirror-tiled, scaled, clamped, centered, cscaled
background_image_linear no
background_tint 0.5
```

9.6 Dim Text

```
# Atenuar texto con atributo DIM
dim_opacity 0.4
```

10 Cursor y Scrollback

10.1 Cursor

Forma:

```
cursor_shape block
# Opciones: block, beam, underline
```

Forma cuando no enfocado:

```
cursor_shape_unfocused hollow
# Opciones: block, beam, underline, hollow, unchanged
```

Grosor:

```
cursor_beam_thickness 1.5
cursor_underline_thickness 2.0
```

Blink:

```
# Intervalo de blink (segundos)
cursor_blink_interval -1
# -1 = usar default del sistema
# 0 = deshabilitado
# >0 = intervalo personalizado

# Parar blink después de inactividad
cursor_stop_blinking_after 15.0

# Animación de blink
cursor_blink_interval 0.5 ease-in-out
```

Color:

```
cursor #cccccc
cursor_text_color #111111
# 0 usar: cursor_text_color background
```

Cursor Trail:

```
# Animación de "estela" del cursor
cursor_trail 5
cursor_trail_decay 0.1 0.4
cursor_trail_start_threshold 2
cursor_trail_color #ff0000
```

10.2 Scrollback

Líneas de historial:

```
scrollback_lines 10000
# -1 = infinito (no recomendado)
```

Pager:

```
scrollback_pager less --chop-long-lines --RAW-CONTROL-CHARS +INPUT_LINE_NUMBER

# Con neovim
scrollback_pager nvim --cmd 'set eventignore=FileType' +'nnoremap q ZQ' +'call nvim_o
```

Scrollback Pager History:

```
# Scrollback separado para pager (MB)
scrollback_pager_history_size 10
```

Fill Window:

```
# Llenar con scrollback al agrandar window
scrollback_fill_enlarged_window no
```

Scroll Multiplier:

```
# Mouse wheel
wheel_scroll_multiplier 5.0
wheel_scroll_min_lines 1

# Touch
touch_scroll_multiplier 1.0
```

10.3 Scrollbar

```
# Cuando mostrar scrollbar
scrollbar scrolled
# Opciones: scrolled, hovered, scrolled-and-hovered, always, never

# Interactivo
scrollbar_interactive yes

# Jump on click
scrollbar_jump_on_click yes

# Ancho
scrollbar_width 0.5
scrollbar_hover_width 1

# Opacidad
scrollbar_handle_opacity 0.5
scrollbar_track_opacity 0

# Colores
scrollbar_handle_color foreground
scrollbar_track_color foreground
```

11 Mouse**11.1 Comportamiento Básico**

```
# Ocultar cursor después de inactividad
mouse_hide_wait 3.0
# 0 = nunca
# <0 = ocultar inmediatamente al escribir

# Focus follows mouse
focus_follows_mouse no
```

11.2 URLs

```
# Color y estilo de URLs
url_color #0087bd
url_style curly
# Opciones: none, straight, double, curly, dotted, dashed

# Programa para abrir URLs
open_url_with default

# Prefijos detectados
url_prefixes file ftp ftps gopher git gopher http https irc ircs kitty mailto news sf

# Detectar URLs
detect_urls yes

# Mostrar target de hyperlinks
show_hyperlink_targets no

# Underline de hyperlinks
underline_hyperlinks hover
# Opciones: hover, always, never
```

11.3 Selección

```
# Copiar al seleccionar
copy_on_select no
# Opciones: no, clipboard, a1 (buffer personalizado)

# Limpiar selección al perder clipboard
clear_selection_on_clipboard_loss no

# Caracteres de palabra
select_by_word_characters @-./_~?&=%+#

# Intervalo de clicks
click_interval -1.0
# -1 = usar default del sistema
```

11.4 Pointer Shape

```
# Forma del cursor cuando programa agarra mouse
pointer_shape_when_grabbed arrow

# Forma default
default_pointer_shape beam

# Forma al arrastrar
pointer_shape_when_dragging beam crosshair
```

11.5 Mouse Actions

Sintaxis:

```
mouse_map button-name event-type modes action
```

Ejemplos:

```
# Click en URL
mouse_map left click ungrabbed mouse_handle_click selection link prompt

# Pegar con middle click
mouse_map middle release ungrabbed paste_from_selection

# Seleccionar con left press
mouse_map left press ungrabbed mouse_selection normal

# Selección rectangular con Ctrl+Alt
mouse_map ctrl+alt+left press ungrabbed mouse_selection rectangle

# Seleccionar palabra con double click
mouse_map left doublepress ungrabbed mouse_selection word

# Seleccionar línea con triple click
mouse_map left triplepress ungrabbed mouse_selection line

# Extender selección con right press
mouse_map right press ungrabbed mouse_selection extend
```

Clearar todos los mouse actions:

```
clear_all_mouse_actions yes
```

12 Kittens (Extensiones)

12.1 ¿Qué son los Kittens?

Programas auxiliares que extienden funcionalidad de kitty.

12.2 Kittens Built-in

12.3 1. icat - Imágenes en Terminal

```
# Mostrar imagen
kitten icat imagen.png

# Con opciones
kitten icat --align left imagen.png
kitten icat --place 40x40@10x10 imagen.png
```

En kitty.conf:

```
map f1 launch --stdin-source=@screen_scrollback kitten icat
```

12.4 2. diff - Diff con Imágenes

```
# Diff de archivos
kitten diff archivo1.txt archivo2.txt

# Con imágenes
kitten diff imagen1.png imagen2.png

# Diff de directorios
kitten diff dir1/ dir2/
```

Configurar como git difftool:

```
git config --global diff.tool kitty
git config --global difftool.kitty.cmd 'kitten diff $LOCAL $REMOTE'
```

12.5 3. unicode_input - Input Unicode

```
# Abrir selector Unicode
Ctrl+Shift+U

# 0
kitten unicode_input
```

12.6 4. themes - Cambiar Temas

```
# Browser de temas
kitten themes

# Listar temas
kitten themes --list
```

```
# Cambiar tema
kitten themes --reload-in=all Dracula
```

12.7 5. choose-fonts - Elegir Fuentes

```
# UI para elegir fuentes
kitten choose-fonts

# Con preview
kitten choose-fonts --preview
```

12.8 6. hints - Selección Rápida

```
# Abrir URLs
Ctrl+Shift+E

# 0 personalizado:
# URLs
kitten hints

# Paths
kitten hints --type path

# Líneas
kitten hints --type line

# Palabras
kitten hints --type word

# Hashes (git)
kitten hints --type hash

# Números de línea
kitten hints --type linenum

# Hyperlinks
kitten hints --type hyperlink
```

Configuración:

```
# Abrir URL
map ctrl+shift+e open_url_with_hints

# Insertar path
map ctrl+shift+p>f kitten hints --type path --program -
```

```
# Abrir path
map ctrl+shift+p>shift+f kitten hints --type path

# Insertar archivo (con chooser)
map ctrl+shift+p>c kitten choose-files

# Insertar directorio
map ctrl+shift+p>d kitten choose-files --mode=dir

# Insertar línea
map ctrl+shift+p>l kitten hints --type line --program -

# Insertar palabra
map ctrl+shift+p>w kitten hints --type word --program -

# Insertar hash
map ctrl+shift+p>h kitten hints --type hash --program -

# Abrir en editor
map ctrl+shift+p>n kitten hints --type linenum

# Abrir hyperlink
map ctrl+shift+p>y kitten hints --type hyperlink
```

12.9 7. ssh - SSH Mejorado

```
# SSH con shell integration
kitten ssh usuario@servidor

# Con forwarding
kitten ssh -R 52698:localhost:52698 servidor
```

Ventajas: - Shell integration automática - Re-uso de conexiones - Clonación de config local - Transferencia de terminfo

12.10 8. transfer - Transferencia de Archivos

```
# Upload archivo
kitten transfer archivo.txt servidor:/ruta/

# Download archivo
kitten transfer servidor:/ruta/archivo.txt ./

# Directorio completo
kitten transfer --recursive directorio/ servidor:/ruta/
```

12.11 9. clipboard - Clipboard en SSH

```
# Copiar a clipboard
echo "texto" | kitten clipboard

# Pegar desde clipboard
kitten clipboard --get-clipboard
```

12.12 10. broadcast - Broadcast Input

```
# Escribir en múltiples windows
kitten broadcast

# A windows específicas
kitten broadcast --match-tab "title:server*"
```

12.13 11. notify - Notificaciones

```
# Enviar notificación
kitten notify "Título" "Mensaje"

# Con icono
kitten notify --icon /path/icon.png "Título" "Mensaje"
```

12.14 12. panel - Panel Desktop

```
# Crear panel
kitten panel sh -c 'while true; do date; sleep 1; done'
```

12.15 13. @ (Remote Control)

```
# Ver todos los comandos
kitten @ --help

# Listar windows
kitten @ ls

# Cambiar colores
kitten @ set-colors foreground=#ffffff

# Ejecutar comando en window
kitten @ send-text --match "title:vim" ":wq\n"
```

12.16 Crear Kittens Personalizados

Estructura básica:

```
#!/usr/bin/env python3
# mi_kitten.py

from kittens.tui.handler import Handler
from kitty.key_encoding import KeyEvent

class MyHandler(Handler):
    def initialize(self):
        self.print("¡Hola desde mi kitten!")

    def on_key(self, key_event: KeyEvent):
        if key_event.key == 'q':
            self.quit_loop(0)
        return True

def main(args):
    handler = MyHandler()
    handler.run()

if __name__ == '__main__':
    import sys
    main(sys.argv[1:])
```

Usar:

```
kitten @ run-kitten /path/to/mi_kitten.py
```

13 Control Remoto

13.1 Habilitar Remote Control

```
# En kitty.conf
allow_remote_control yes

# Socket Unix
listen_on unix:/tmp/kitty

# O TCP
listen_on tcp:localhost:12345
```

13.2 Uso Básico

```
# Listar windows
kitty @ ls

# Crear nueva window
kitty @ launch

# Cerrar window
kitty @ close-window

# Cambiar título
kitty @ set-window-title "Nuevo Título"

# Enviar texto
kitty @ send-text "echo hello\n"
```

13.3 Comandos Principales

Información:

```
kitty @ ls          # Listar todo
kitty @ get-colors  # Ver colores actuales
kitty @ get-text    # Obtener texto de window
```

Crear/Cerrar:

```
kitty @ launch      # Nueva window
kitty @ launch --type=tab  # Nueva tab
kitty @ launch --type=os-window # Nueva OS window
kitty @ close-window
kitty @ close-tab
```

Modificar:

```
kitty @ set-colors foreground=#ff0000
kitty @ set-window-title "Título"
kitty @ set-tab-title "Tab"
kitty @ resize-window --increment 10
kitty @ goto-layout tall
```

Enviar Input:

```
kitty @ send-text "texto"
kitty @ send-text --match "title:vim" ":wq\n"
```

Focus:

```
kitty @ focus-window --match "title:editor"
kitty @ focus-tab --match "index:2"
```

13.4 Matching

Usar `--match` para seleccionar windows/tabs:

```
# Por título
--match "title:editor"

# Por ID
--match "id:42"

# Por estado
--match "state:focused"

# Múltiples condiciones
--match "title:vim and state:focused"
```

13.5 Remote Control desde Mapeos

```
# Crear window con comando específico
map f1 remote_control launch --type=tab vim

# Cambiar colores
map f2 remote_control set-colors --configured ~/colors.conf

# Broadcast
map f3 remote_control send-text --all "clear\n"
```

13.6 Remote Control via SSH

```
# Desde máquina remota
export KITTY_REMOTE_CONTROL_PASSWORD="mi_password"
kitty @ --password=$KITTY_REMOTE_CONTROL_PASSWORD ls
```

Con kitten ssh:

```
# Automáticamente habilita remote control
kitten ssh servidor

# Desde el servidor
kitty @ ls # ¡Funciona!
```

14 Sesiones

14.1 ¿Qué son las Sesiones?

Archivos que definen:

- Layout de tabs y windows
- Directorio de trabajo
- Comandos a ejecutar
- Configuración específica

14.2 Crear Sesión

Archivo de sesión:

```
# mi_sesion.kitty

# Nueva ventana
new_window
cd ~/proyectos/app
launch vim

# Nueva ventana
new_window
cd ~/proyectos/app
launch

# Nueva tab
new_tab editor
cd ~/proyectos/app
layout tall
launch vim src/main.py
launch

# Nueva tab con split
new_tab servers
launch ssh server1
launch --location=hsplit ssh server2
```

14.3 Usar Sesión

Al iniciar kitty:

```
kitty --session mi_sesion.kitty
```

Desde configuración:

```
startup_session mi_sesion.kitty
```

Desde atajo:

```
map f1 launch --type=os-window --session=mi_sesion.kitty
```

14.4 Comandos de Sesión

new_tab:

```
new_tab [nombre]  
cd /directorio  
layout tall  
launch comando
```

new_window:

```
new_window  
cd /directorio  
launch comando
```

launch:

```
launch [--location=split] comando
```

cd:

```
cd /ruta/absoluta  
cd ~/relativa
```

layout:

```
layout tall  
layout stack  
layout splits
```

focus:

```
focus
```

enabled_layouts:

```
enabled_layouts tall,stack,grid
```

14.5 Ejemplo Completo

```
# Sesión de desarrollo web
# ~/proyectos/web_dev.kitty

# Tab principal - Editor
new_tab Editor
cd ~/proyectos/webapp
layout tall:bias=70;full_size=1
launch nvim .

# Tab de servidores
new_tab Servers
cd ~/proyectos/webapp
layout splits
launch npm run dev
launch --location=hsplit npm run tailwind

# Tab de terminal
new_tab Terminal
cd ~/proyectos/webapp
launch

# Tab de git
new_tab Git
cd ~/proyectos/webapp
launch lazygit

# Focus en primera tab
focus_tab 1
```

14.6 Guardar Sesión Actual

Kitty no tiene comando built-in para guardar, pero puedes usar remote control:

```
# Script para guardar sesión
kitty @ ls | python3 crear_sesion.py > sesion_actual.kitty
```

14.7 Switching Sessions

Con goto_session:

```
# Crear/cambiar a sesión
map f1 goto_session dev
map f2 goto_session personal
```

Cerrar sesión:

```
# Cerrar todas las windows de una sesión
map f3 close_session dev
```

15 Integración con Shell

15.1 Shell Integration

Kitty puede integrarse con bash, zsh y fish para:

- Saltar a prompts anteriores
- Ver output de comandos
- Tracking de directorios
- Indicadores visuales

15.2 Habilitar

```
shell_integration enabled
```

Deshabilitar características específicas:

```
# Deshabilitar cursor shape en prompt
shell_integration no-cursor

# Deshabilitar título automático
shell_integration no-title

# Deshabilitar tracking de cwd
shell_integration no-cwd

# Deshabilitar prompt marks
shell_integration no-prompt-mark

# Deshabilitar completion
shell_integration no-complete

# Deshabilitar sudo
shell_integration no-sudo
```

15.3 Configuración Manual

Si la integración automática no funciona:

Bash (~/.bashrc):

```
if [[ -n "$KITTY_INSTALLATION_DIR" ]]; then
    export KITTY_SHELL_INTEGRATION="enabled"
    source "$KITTY_INSTALLATION_DIR/shell-integration/bash/kitty.bash"
fi
```

Zsh (~/.zshrc):

```
if [[ -n "$KITTY_INSTALLATION_DIR" ]]; then
    export KITTY_SHELL_INTEGRATION="enabled"
    source "$KITTY_INSTALLATION_DIR/shell-integration/zsh/kitty.zsh"
fi
```

Fish (~/.config/fish/config.fish):

```
if set -q KITTY_INSTALLATION_DIR
    set --global KITTY_SHELL_INTEGRATION enabled
    source "$KITTY_INSTALLATION_DIR/shell-integration/fish/vendor_conf.d/kitty-shell-
    set --prepend fish_complete_path "$KITTY_INSTALLATION_DIR/shell-integration/fish/
end
```

15.4 Características con Shell Integration

Saltar a prompts:

```
Ctrl+Shift+Z # Prompt anterior
Ctrl+Shift+X # Prompt siguiente
```

Ver output de comandos:

```
Ctrl+Shift+G # Ver último comando en pager
```

Mouse:

- Click en prompt para mover cursor
- Right-click en comando para ver output
- Click en error para copiar

Indicadores visuales:

- Marca en prompt
- Color diferente para comandos exitosos/fallidos

16 Marcas (Marks)

16.1 ¿Qué son las Marcas?

Sistema para resaltar texto basado en regex.

16.2 Tipos de Marcas

Hay 3 tipos (cada uno con color diferente):

- mark1 (light steel blue)
- mark2 (beige)
- mark3 (violet)

16.3 Configurar Colores

```
# Mark tipo 1
mark1_foreground black
mark1_background #98d3cb

# Mark tipo 2
mark2_foreground black
mark2_background #f2dcd3

# Mark tipo 3
mark3_foreground black
mark3_background #f274bc
```

16.4 Crear Marcas

Con regex:

```
# Marcar URLs
map f1 create_marker text 1 https?://[^\s]+

# Marcar errores
map f2 create_marker text 1 ERROR|FATAL

# Marcar TODO
map f3 create_marker text 2 TODO|FIXME|XXX
```

Desde línea de comando:

```
# Marcar palabra
kitty @ create-marker text 1 "error"

# Marcar líneas que contengan
kitty @ create-marker iregex 1 "^error.*"
```

16.5 Usar Marcas

Toggle:

```
map f1 toggle_marker text 1 error
```

Saltar:

```
# Siguiente marca
map f2 scroll_to_mark next 1

# Anterior marca
map f3 scroll_to_mark prev 1

# Cualquier tipo
map f4 scroll_to_mark next 0
```

Eliminar:

```
map f5 remove_marker
```

16.6 Ejemplo Completo

```
# Marcar diferentes niveles de log
map ctrl+shift+1 create_marker iregex 1 "^.*ERROR.*$"
map ctrl+shift+2 create_marker iregex 2 "^.*WARNING.*$"
map ctrl+shift+3 create_marker iregex 3 "^.*INFO.*$"

# Navegar entre marcas
map ctrl+shift+n scroll_to_mark next
map ctrl+shift+p scroll_to_mark prev

# Toggle marcas
map ctrl+shift+m toggle_marker iregex 1 "^.*ERROR.*$"

# Limpiar todas las marcas
map ctrl+shift+c remove_marker
```

17 Características Avanzadas**17.1 Multiple Copy/Paste Buffers**

```
# Definir buffer 'a'
map f1 copy_to_buffer a
map f2 paste_from_buffer a

# Definir buffer 'b'
map f3 copy_to_buffer b
map f4 paste_from_buffer b
```

17.2 Keyboard Modes

Modos de teclado para mapeos contextuales:

```
# Crear modo
map f1 push_keyboard_mode resize

# En modo resize
map --mode resize h resize_window narrower
map --mode resize l resize_window wider
map --mode resize k resize_window taller
map --mode resize j resize_window shorter
map --mode resize esc pop_keyboard_mode
```

17.3 Per-Application Mappings

```
# Solo para vim
map --when-focus-on vim ctrl+j send_text \x0a

# Solo para less
map --when-focus-on less j send_text j
```

17.4 Environment Variables

```
# Establecer variables
env VAR=value
env PATH=/custom/path:${PATH}

# Leer de shell
env read_from_shell=PATH LANG LC_* XDG_*
```

17.5 Watchers

Scripts Python que reaccionan a eventos:
~/.config/kitty/watchers.py:

```
from kitty.boss import Boss

def on_focus_change(boss: Boss, window, data):
    if data['focused']:
        # Window ganó focus
        pass
    else:
        # Window perdió focus
        pass

def on_close(boss: Boss, window, data):
    # Window está cerrándose
    pass

def on_resize(boss: Boss, window, data):
    # Window fue redimensionada
    pass

def on_title_change(boss: Boss, window, data):
    new_title = data['title']
    # Hacer algo con nuevo título
    pass
```

Habilitar:

```
watcher ~/.config/kitty/watchers.py
```

17.6 Filtros de Notificación

```
# Filtrar notificaciones por campo
filter_notification title:hello or body:"abc.*def"
filter_notification app:"[ng]?vim" and not body:"(?i)update"

# Filtrar todas
filter_notification all
```

17.7 Allow Cloning

```
# Permitir clone-in-kitty
allow_cloning ask
# Opciones: yes, no, ask
```

17.8 Notificaciones de Comando

```
# Notificar cuando comando termina
notify_on_cmd_finish unfocused 10.0

# Con bell
notify_on_cmd_finish invisible 10.0 bell

# Con comando personalizado
notify_on_cmd_finish invisible 10.0 command notify-send "Done" "%c"
```

18 Optimización de Rendimiento

18.1 Repaint Delay

```
# Delay entre actualizaciones (ms)
repaint_delay 10
# Menor = más FPS pero más CPU
```

18.2 Input Delay

```
# Delay antes de procesar input (ms)
input_delay 3
# Menor = más responsive pero más CPU
```

18.3 Sync to Monitor

```
# Sincronizar con refresh rate
sync_to_monitor yes
# no = más FPS pero posible tearing
```

18.4 Scrollback

```
# No mantener scrollback enorme
scrollback_lines 10000

# Usar scrollback_pager_history_size
scrollback_pager_history_size 10
```

18.5 Text Composition

```
# Estrategia de composición
text_composition_strategy platform
# platform, legacy, o números para ajuste fino
```

18.6 OpenGL

```
# Verificar que usa GPU
glxinfo | grep "OpenGL renderer"

# Actualizar drivers GPU
sudo apt update && sudo apt upgrade
```

19 Trucos y Tips

19.1 Cambiar Directorio del Shell al Salir

Bash/Zsh:

```
# Añadir a ~/.bashrc o ~/.zshrc
alias icat="kitten icat"
```

19.2 Previsualizar Imágenes

```
# En ranger
set preview_images true
set preview_images_method kitty
```

19.3 Integración con Neovim

```
-- En init.lua
vim.opt.termguicolors = true

-- Copiar a clipboard de kitty
vim.keymap.set('v', '<C-c>', '"+y')
```

19.4 SSH con Túneles

```
# Túnel reverso para remote control
ssh -R 52698:localhost:52698 servidor

# O con kitten ssh (automático)
kitten ssh servidor
```

19.5 Diff de Directorios

```
kitten diff dir1/ dir2/
```

19.6 Broadcast Input

```
# Escribir en múltiples servers
kitten broadcast --match-tab "title:server*"
```

19.7 Quick Terminal (Quake-style)

```
# Crear script kitty-quake.sh
#!/bin/bash
kitty --class=kitty-quake --name quake \
  -o initial_window_width=100c \
  -o initial_window_height=30c \
  -o remember_window_size=no
```

Configurar en i3/sway:

```
for_window [class="kitty-quake"] floating enable, move position center
bindsym $mod+grave exec ~/.local/bin/kitty-quake.sh
```

19.8 8. URLs Personalizadas

```
# Abrir issues de GitHub
map ctrl+shift+o kitten hints --type=url --regex "\\b[A-Z]+-\\d+" --program @
```

19.9 Layouts Personalizados

Crear ~/.config/kitty/layouts/my_layout.py:

```
from kitty.layout import Layout

class MyLayout(Layout):
    name = "my_layout"

    def do_layout(self, windows, active_window_idx):
        # Tu código de layout
        pass
```

19.10 Themevar para Colores Dinámicos

```
# Variables de color
env KITTY_FG=#ffffff
env KITTY_BG=#000000

# Usar en config
foreground ${KITTY_FG}
background ${KITTY_BG}
```

20 Solución de Problemas

20.1 Problemas Comunes

20.2 Ligaduras No Funcionan

Verificar:

```
# Ver features de fuente
fc-list : family fontformat | grep -i "fira code"

# Verificar soporte
kitty +list-fonts --psnames | grep -i "fira"
```

Solución:

```
# Habilitar explícitamente
disable_ligatures never

# O instalar fuente con ligaduras
sudo apt install fonts-firacode
```

20.3 Colores Incorrectos

Verificar:

```
# Verificar TERM
echo $TERM # Debe ser xterm-kitty

# Test de colores
kitty +kitten themes
```

Solución:

```
# Forzar TERM
term xterm-kitty

# O reinstalar terminfo
kitty +kitten terminfo
```

20.4 Imágenes No se Muestran**Verificar:**

```
# Test básico
kitten icat /path/to/image.png
```

Solución:

```
# Verificar OpenGL
glxinfo | grep "OpenGL"

# Actualizar drivers
sudo apt update && sudo apt upgrade
```

20.5 Performance Lento**Diagnóstico:**

```
# Ver stats de rendimiento
kitty --debug-rendering

# Ver FPS
kitty --debug-gl
```

Solución:

```
# Ajustar delays
repaint_delay 10
input_delay 3

# Reducir scrollbar
scrollback_lines 5000

# Deshabilitar features pesadas
cursor_blink_interval 0
```

20.6 Remote Control No Funciona**Verificar:**

```
# Ver si está escuchando
netstat -an | grep kitty

# 0
lsof -i :12345
```

Solución:

```
# Habilitar
allow_remote_control yes
listen_on unix:/tmp/kitty

# Verificar permisos
chmod 600 /tmp/kitty
```

20.7 Fuentes Borrosas**Solución:**

```
# Ajustar composition
text_composition_strategy 1.0 0

# 0 en macOS
text_composition_strategy 1.7 30
```

20.8 Clipboard No Funciona**Verificar:**

```
# Ver proveedores de clipboard
xclip -selection clipboard -o
```

Solución:

```
# Instalar xclip
sudo apt install xclip

# 0 wl-clipboard para Wayland
sudo apt install wl-clipboard
```

20.9 Debugging**Modo verbose:**

```
kitty --debug-config
kitty --debug-rendering
kitty --debug-gl
```

Ver logs:

```
# Linux
~/.local/state/kitty/log

# macOS
~/Library/Logs/kitty
```

21 Referencia Completa

21.1 Archivos de Configuración

```
~/.config/kitty/
  kitty.conf           # Configuración principal
  current-theme.conf   # Tema actual (generado)
  diff.conf            # Config para kitty diff
  sessions/            # Archivos de sesión
    dev.kitty
    personal.kitty
  layouts/              # Layouts personalizados
    my_layout.py
  watchers.py           # Watchers de eventos
```

21.2 Variables de Entorno

```
KITTY_CONFIG_DIRECTORY # Dir de config
KITTY_INSTALLATION_DIR  # Dir de instalación
KITTY_PID                # PID de kitty
KITTY_WINDOW_ID          # ID de window actual
KITTY_LISTEN_ON          # Socket de remote control
```

21.3 Códigos de Escape Útiles

```
# Cambiar título de window
echo -e "\033]2;Nuevo Título\007"

# Cambiar color
echo -e "\033]11;#ff0000\007"

# Mostrar imagen
echo -e "\033]1337;File=inline=1:$(base64 imagen.png)\007"
```

21.4 API de Remote Control

Endpoints principales:

- `ls` - Listar windows/tabs
- `launch` - Crear window/tab
- `close-window` - Cerrar window
- `close-tab` - Cerrar tab

- `focus-window` - Enfocar window
- `focus-tab` - Enfocar tab
- `get-colors` - Obtener colores
- `set-colors` - Cambiar colores
- `get-text` - Obtener texto
- `send-text` - Enviar texto
- `resize-window` - Redimensionar
- `set-window-title` - Cambiar título
- `goto-layout` - Cambiar layout

21.5 Comparison Chart

Característica	Kitty	Alacritty	iTerm2	Wezterm
GPU Rendering				
Tabs				
Splits				
Imágenes				
Ligaduras				
Remote Control				
Extensiones	(kittens)			(lua)
Cross-platform				
Config Language	conf	toml	GUI	lua
Shell Integration				

22 Conclusión

22.1 Roadmap de Aprendizaje

Semana 1 - Básicos:

- Instalar y configurar
- Aprender atajos principales
- Configurar fuentes y colores

Semana 2 - Intermedio:

- Dominar tabs y windows
- Experimentar con layouts
- Configurar shell integration

Mes 1 - Avanzado:

- Usar kittens
- Remote control básico
- Crear sesiones

Mes 2+ - Experto:

- Crear kittens propios
- Watchers
- Automatización completa

22.2 Recursos

Documentación Oficial:

- <https://sw.kovidgoyal.net/kitty/>
- <https://sw.kovidgoyal.net/kitty/conf/>

Comunidad:

- GitHub: <https://github.com/kovidgoyal/kitty>
- Discussions: <https://github.com/kovidgoyal/kitty/discussions>
- Reddit: [r/KittyTerminal](https://www.reddit.com/r/KittyTerminal)

Temas:

- <https://github.com/dexpota/kitty-themes>
- <https://github.com/kdrag0n/base16-kitty>

Configs de Ejemplo:

- <https://github.com/search?q=kitty.conf>

22.3 Tips Finales

1. **Empieza simple:** No copies configs complejas al inicio
2. **Experimenta:** Prueba diferentes layouts y kittens
3. **Lee la docs:** La documentación de kitty es excelente
4. **Usa kitten themes:** Para encontrar tu tema perfecto
5. **Aprovecha shell integration:** Hace la diferencia
6. **Explora kittens:** Son muy potentes
7. **Practica atajos:** La eficiencia viene de no usar el mouse
8. **Contribuye:** Reporta bugs, sugerencias, o crea kittens

23 Publicaciones Similares

Si te interesó este artículo, te recomendamos que explores otros blogs y recursos relacionados que pueden ampliar tus conocimientos. Aquí te dejo algunas sugerencias:

1.  [Comandos De Informacion Windows](#)
2.  [Adb](#)
3.  [Limpieza Y Optimizacion De Pc](#)
4.  [Usando Apk En Windown 11](#)
5.  [Gestionar Versiones De Jdk En Kubuntu](#)
6.  [Instalar Tor Browser](#)
7.  [Crear Enlaces Duros O Hard Link En Linux](#)
8.  [Comandos Vim](#)
9.  [Guia De Git Y Github](#)
10.  [00 Primeros Pasos En Linux](#)
11.  [01 Introduccion Linux](#)
12.  [02 Distribuciones Linux](#)
13.  [03 Instalacion Linux](#)

14.  [04 Administracion Particiones Volúmenes](#)
15.  [Atajos De Teclado Y Comandos Para Usar Vim](#)
16.  [Instalando Specitify](#)
17.  [Gestiona Tus Dotfiles Con Gnu Stow](#)

Esperamos que encuentres estas publicaciones igualmente interesantes y útiles. ¡Disfruta de la lectura!