

# | SpaceShell: Desarrollo de una Shell básica

## Sistemas Operativos - Evaluación de Producto

William Yak Vargas Chambilla    Eduardo Xavier Paca Aquino

Octubre 2025

qn sube el link al aula?pera Desarrollar e implementar **SpaceShell**, un intérprete de comandos con funcionalidades básicas como evidencia de conocimientos en Sistemas Operativos.

## ¿Qué es una Shell?

Interfaz que permite al usuario comunicarse con el Kernel del Sistema Operativo. El intérprete invoca al Kernel, más no lo reemplaza.

### Características clave:

- CLI (Interfaz de Línea de Comandos)
- Interfaz textual sin elementos visuales estructurados
- Similar a BASH en sistemas Unix

# Requisitos Funcionales

Funcionalidades Core

# Requisitos Funcionales (1/3)

## 1. Prompt personalizado

- Mostrar prompt propio y leer línea de comando

## 2. Resolución de rutas

- Rutas absolutas se ejecutan tal cual
- Si no es absoluta, se asume /bin
- Manejo de errores de permisos/existencia

## 3. Ejecución mediante procesos

- Invocación con `fork()` y `exec()`
- Proceso padre espera con `wait()/waitpid()`

# Requisitos Funcionales (2/3)

## 4. Manejo de Errores

- Mensajes claros cuando el comando no existe
- Uso de errno/perror para diagnósticos

## 5. Redirección de salida (>)

- Redirigir stdout a archivo
- Crear/truncar archivo destino

## 6. Comando de salida

- Terminar con salir o Ctrl + D

# Requisitos Funcionales (3/3)

## 7. Pipes

- Ejecutar pipes simples: `cmd1 | cmd2`

## 8. Tareas en segundo plano

- Usar ampersand & para no bloquear prompt

## 9. Redirección avanzada

- Entrada < y salida append >>

## 10. Comandos internos (Built-ins)

- Comandos como `cd`, `pwd`, `help`, `history`

## 11. Concurrencia con hilos

■ Built-in PARALLEL que ejecute n comandos en paralelo con pthread\_create

## 12. Gestión de memoria instrumentada

- Built-in meminfo mostrando uso de heap
- Estadísticas de asignaciones/liberaciones

## 13. Manejo de señales

- Capturar SIGINT en el padre
- Pasar señales a hijos cuando corresponda

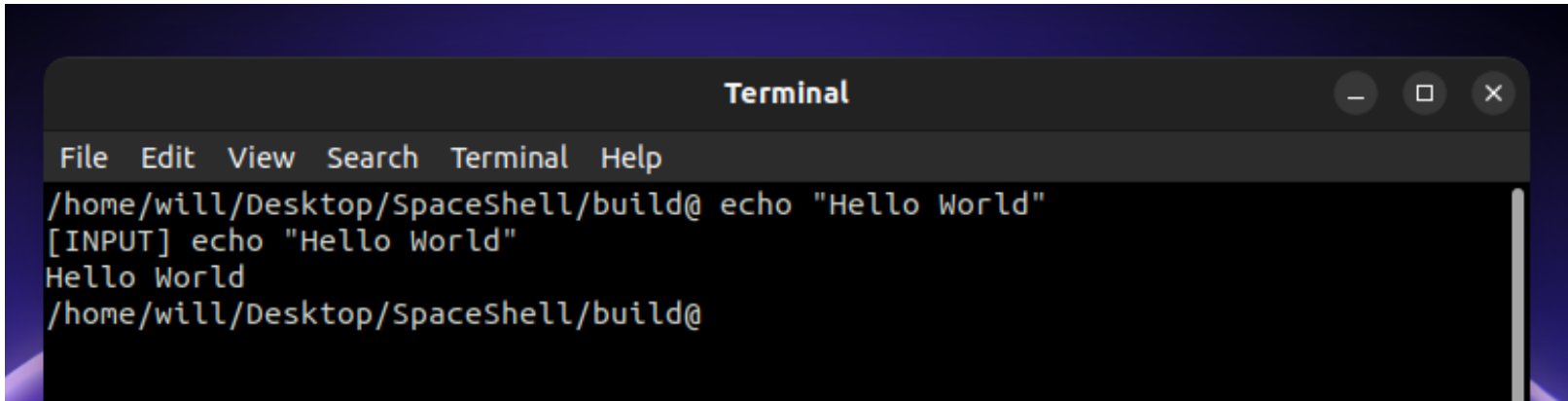
# Pruebas y Resultados

Demostración de Funcionalidades



# Prompt Personalizado

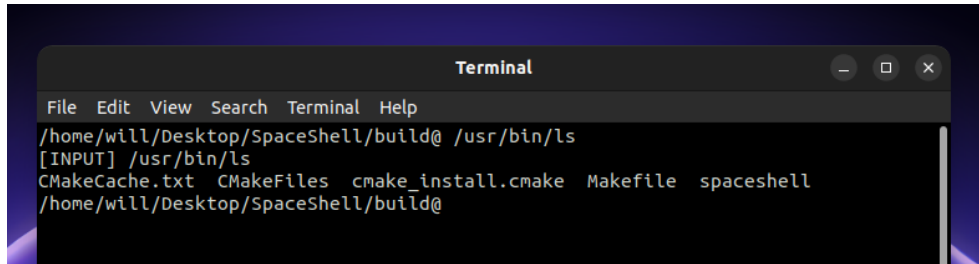
El prompt muestra el directorio actual seguido del símbolo @.

A screenshot of a terminal window titled "Terminal". The window has a dark background and a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal shows a custom prompt: `/home/will/Desktop/SpaceShell/build@`. The user has entered `echo "Hello World"`, and the terminal has outputted `Hello World`. The prompt is followed by a cursor.

```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ echo "Hello World"
[INPUT] echo "Hello World"
Hello World
/home/will/Desktop/SpaceShell/build@
```

Proporciona contexto visual sobre la ubicación en el sistema de archivos.

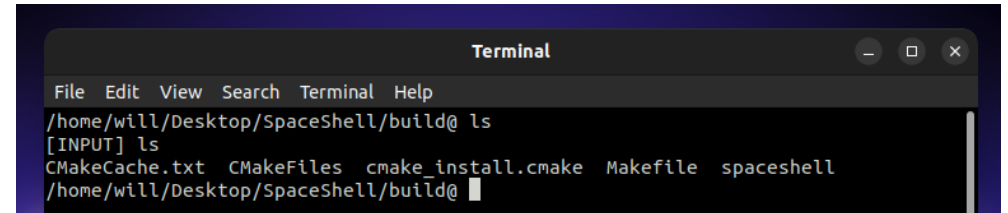
## Rutas Absolutas



```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ /usr/bin/ls
[INPUT] /usr/bin/ls
CMakeCache.txt CMakeFiles cmake_install.cmake Makefile spaceshell
/home/will/Desktop/SpaceShell/build@
```

Comienzan con /, se ejecutan directamente

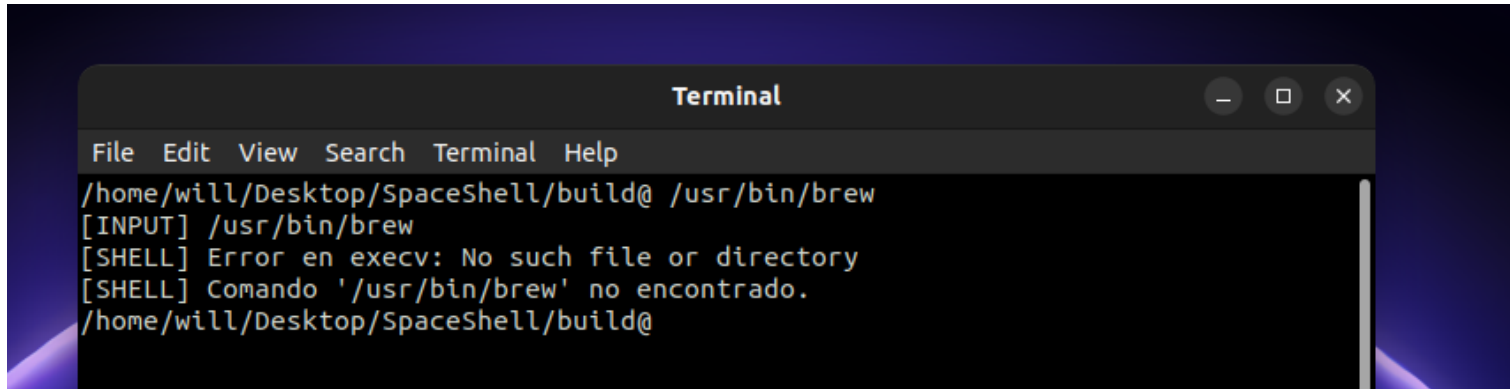
## Rutas Relativas



```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ ls
[INPUT] ls
CMakeCache.txt CMakeFiles cmake_install.cmake Makefile spaceshell
/home/will/Desktop/SpaceShell/build@
```

Se antepone /bin/ automáticamente

Mensajes claros cuando un comando no puede ejecutarse.

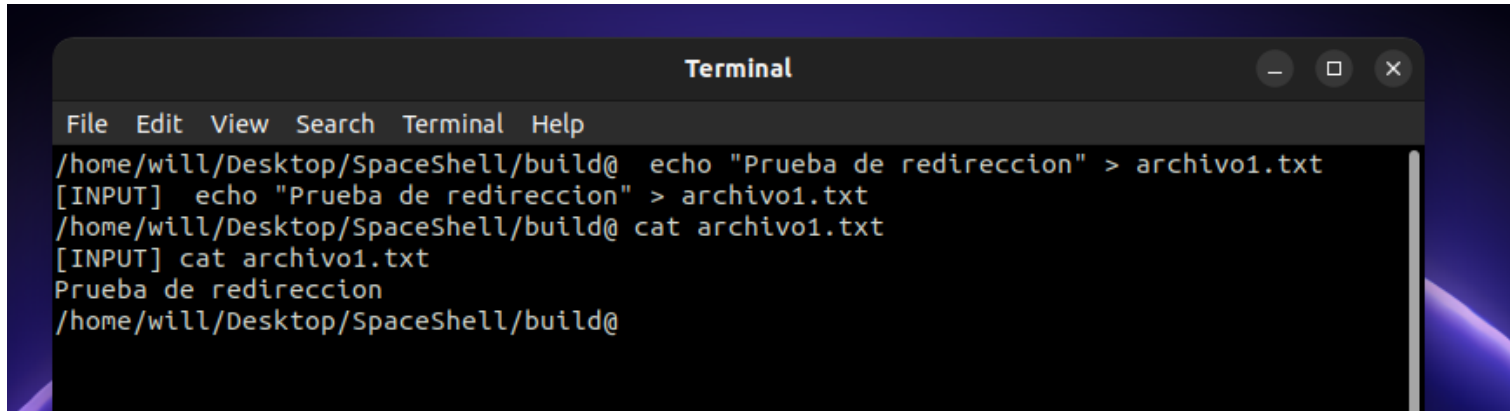


```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ /usr/bin/brew
[INPUT] /usr/bin/brew
[SHELL] Error en execv: No such file or directory
[SHELL] Comando '/usr/bin/brew' no encontrado.
/home/will/Desktop/SpaceShell/build@
```

Usa `perror()` y `errno` para diagnósticos precisos del sistema.

# Redirección de Salida

Operador > redirige salida estándar a archivo.

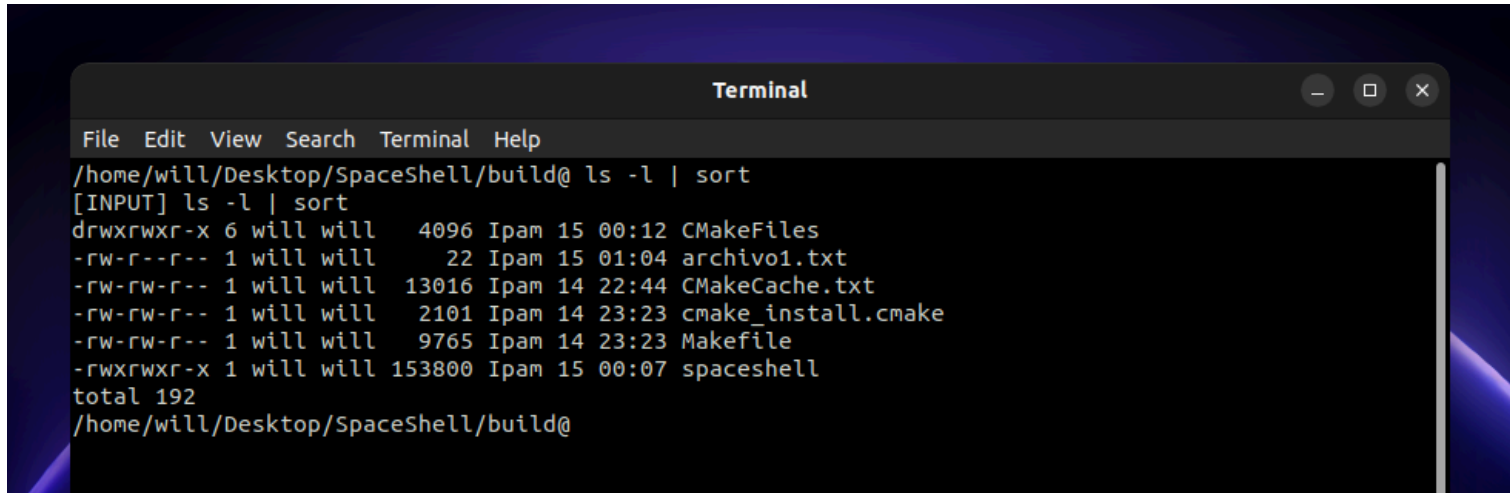
A screenshot of a terminal window titled "Terminal" with standard window controls (minimize, maximize, close). The terminal has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The command prompt is "/home/will/Desktop/SpaceShell/build@". The user enters the command "echo \"Prueba de redireccion\" > archivo1.txt". The terminal shows "[INPUT] echo \"Prueba de redireccion\" > archivo1.txt". The user then enters "cat archivo1.txt". The terminal shows "[INPUT] cat archivo1.txt" followed by the output "Prueba de redireccion". The prompt returns to "/home/will/Desktop/SpaceShell/build@".

```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ echo "Prueba de redireccion" > archivo1.txt
[INPUT] echo "Prueba de redireccion" > archivo1.txt
/home/will/Desktop/SpaceShell/build@ cat archivo1.txt
[INPUT] cat archivo1.txt
Prueba de redireccion
/home/will/Desktop/SpaceShell/build@
```

Trunca si existe o crea nuevo archivo.

# Pipes (Tuberías)

`cmd1 | cmd2` conecta salida de un comando con entrada del siguiente.

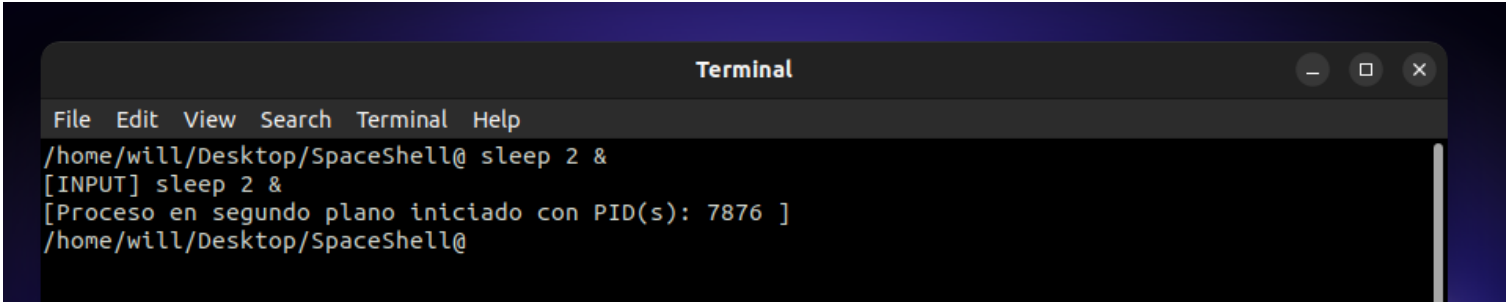
A screenshot of a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal shows the command `/home/will/Desktop/SpaceShell/build@ ls -l | sort` being executed. The output is a sorted list of files and directories. The prompt `[INPUT]` is shown before the command is entered. The output lines are: `drwxrwxr-x 6 will will 4096 Ipam 15 00:12 CMakeFiles`, `-rw-r--r-- 1 will will 22 Ipam 15 01:04 archivo1.txt`, `-rw-rw-r-- 1 will will 13016 Ipam 14 22:44 CMakeCache.txt`, `-rw-rw-r-- 1 will will 2101 Ipam 14 23:23 cmake_install.cmake`, `-rw-rw-r-- 1 will will 9765 Ipam 14 23:23 Makefile`, `-rwxrwxr-x 1 will will 153800 Ipam 15 00:07 spaceshell`, and `total 192`. The prompt `/home/will/Desktop/SpaceShell/build@` is shown at the bottom.

```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ ls -l | sort
[INPUT] ls -l | sort
drwxrwxr-x 6 will will 4096 Ipam 15 00:12 CMakeFiles
-rw-r--r-- 1 will will 22 Ipam 15 01:04 archivo1.txt
-rw-rw-r-- 1 will will 13016 Ipam 14 22:44 CMakeCache.txt
-rw-rw-r-- 1 will will 2101 Ipam 14 23:23 cmake_install.cmake
-rw-rw-r-- 1 will will 9765 Ipam 14 23:23 Makefile
-rwxrwxr-x 1 will will 153800 Ipam 15 00:07 spaceshell
total 192
/home/will/Desktop/SpaceShell/build@
```

Permite crear tuberías de procesamiento de datos.

# Tareas en Segundo Plano

Operador & ejecuta comandos sin bloquear el prompt.

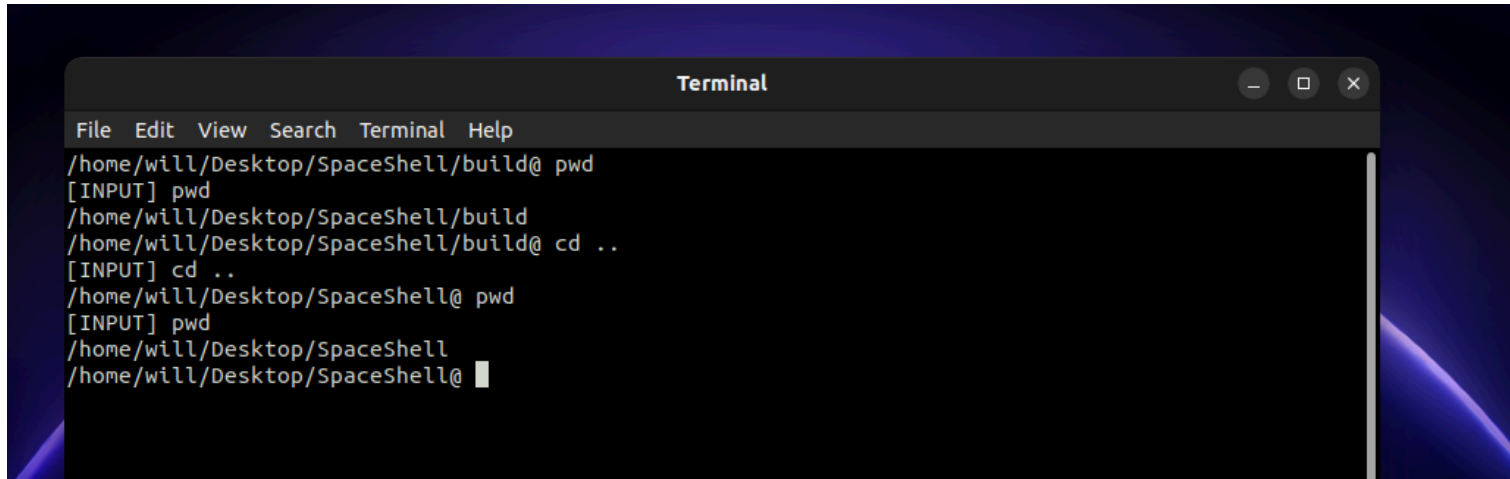
A screenshot of a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the following sequence of commands and outputs:

```
/home/will/Desktop/SpaceShell@ sleep 2 &  
[INPUT] sleep 2 &  
[Proceso en segundo plano iniciado con PID(s): 7876 ]  
/home/will/Desktop/SpaceShell@
```

Muestra PIDs de los procesos iniciados.

# Comandos Internos (Built-ins)

Se ejecutan en el proceso de la shell sin crear procesos hijos.

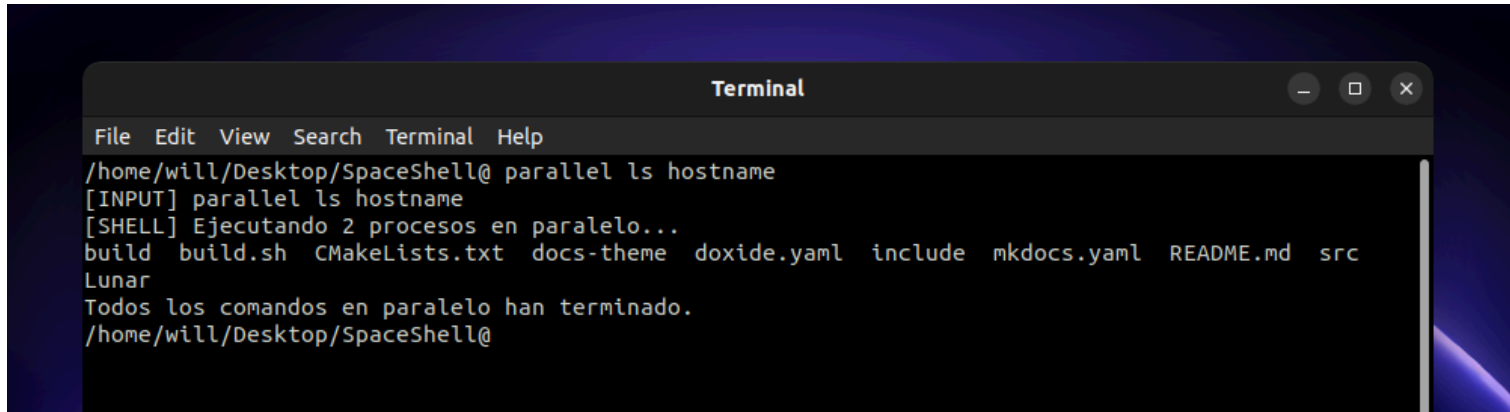
A screenshot of a terminal window titled "Terminal" with standard macOS window controls. The terminal shows a sequence of built-in commands and their outputs. The prompt is "/home/will/Desktop/SpaceShell/build@". The user enters "pwd", and the output is "/home/will/Desktop/SpaceShell/build". Then the user enters "cd ..", and the output is "/home/will/Desktop/SpaceShell@". Finally, the user enters "pwd" again, and the output is "/home/will/Desktop/SpaceShell". The terminal has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help".

```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell/build@ pwd
[INPUT] pwd
/home/will/Desktop/SpaceShell/build
/home/will/Desktop/SpaceShell/build@ cd ..
[INPUT] cd ..
/home/will/Desktop/SpaceShell@ pwd
[INPUT] pwd
/home/will/Desktop/SpaceShell
/home/will/Desktop/SpaceShell@
```

Permiten modificar el estado del intérprete (cd, pwd).

# Ejecución Paralela

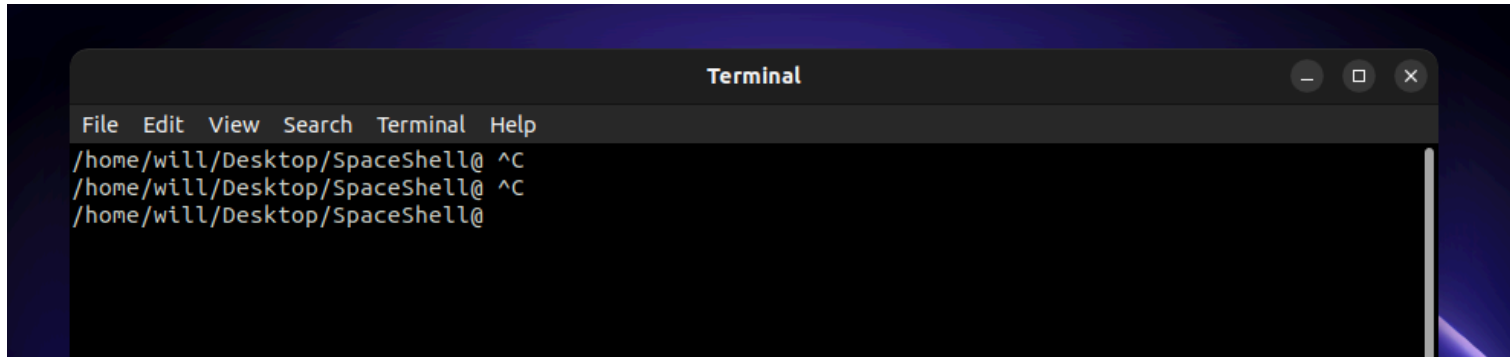
Comando `parallel` ejecuta múltiples comandos concurrentemente.

A screenshot of a terminal window titled "Terminal" with standard window controls (minimize, maximize, close). The terminal has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The prompt is `/home/will/Desktop/SpaceShell@`. The user enters `parallel ls hostname`. The terminal shows the input, then a message: `[SHELL] Ejecutando 2 procesos en paralelo...`. Below this, two lines of output are shown: `build build.sh CMakeLists.txt docs-theme doxide.yaml include mkdocs.yaml README.md src` and `Lunar`. A status message follows: `Todos los comandos en paralelo han terminado.`. The prompt returns to `/home/will/Desktop/SpaceShell@`.

Usa `pthread_create` y sincroniza con `pthread_join`.



Captura SIGINT (Ctrl+C) sin terminar la shell.



```
Terminal
File Edit View Search Terminal Help
/home/will/Desktop/SpaceShell@ ^C
/home/will/Desktop/SpaceShell@ ^C
/home/will/Desktop/SpaceShell@
```

Los procesos hijos pueden recibir la señal normalmente.

## Logros

- Shell funcional con características POSIX
- Diseño modular y escalable
- Interacción con librerías de bajo nivel
- Implementación de concurrencia y paralelismo

## Trabajos Futuros

- Menús detallados y personalización de interfaces
- Manejo de niveles de usuario (Seguridad)
- Funcionalidades avanzadas de usuario

¡Gracias!

UNJBG - Tacna, Perú ■



**William Yak Vargas Chambilla**

2023-119066

**Eduardo Xavier Paca Aquino**

2023-119049

Sistemas Operativos - Ciclo VI  
MSc. Hugo Manuel Barraza Vizcarra