

Informe de Proyecto - Tarea 5

Sistemas Operativos

Cristian Olarte
Ana Vargas
Universidad Santo Tomas

Mayo 2025

Resumen

Este informe presenta el desarrollo de la Tarea 5 del curso de Sistemas Operativos, que comprende tres componentes clave: un simulador de carro seguidor de línea con control PID, un videojuego personalizado tipo "space shooter" y un sistema básico de comunicación en ROS, todo ejecutado sobre contenedores Docker. Se detalla el funcionamiento, estructura y ejecución de cada módulo, así como reflexiones finales del proceso.

1 Introducción

La Tarea 5 busca integrar conocimientos de programación, simulación, sistemas gráficos y contenedores. El objetivo es crear soluciones prácticas que demuestren interacción con el sistema operativo mediante la ejecución de aplicaciones gráficas en contenedores, manejo de procesos y comunicación.

2 Carro Seguidor de Línea

Tecnologías: Tkinter, PID, Docker

Este módulo simula un vehículo autónomo que sigue una pista con curvas. Emula sensores laterales y aplica un controlador PID para ajustar su trayectoria dinámicamente.

Características Principales

- Interfaz gráfica creada con Tkinter
- Control PID ajustable
- Pista curva personalizada
- Contenedor Docker con entorno gráfico

Estructura del Proyecto

```
1 carro_seguidor_docker/  
2     CarroFinal.py  
3     Dockerfile  
4     Video De Carrito Seguidor.mkv  
5     README.md
```

Ejecución

Local:

```
1 python CarroFinal.py
```

Docker:

```
1 docker build -t carro_gui .  
2  
3 xhost +local:root  
4 docker run -it \  
5     -e DISPLAY=$DISPLAY \  
6     -v /tmp/.X11-unix:/tmp/.X11-unix \  
7     --rm carro_gui
```

Docker Hub:

```
1 docker push cristian039/carro-seguidor:latest  
2 docker pull cristian039/carro-seguidor:latest
```

3 Galaxy Rush – Videojuego Espacial

Tecnologías: Pygame, Docker

Se personalizó un juego tipo arcade con sprites y sonidos únicos, mecánicas de jefe final y power-ups.

Componentes del Juego

- Power-ups: velocidad, disparo doble y rápido
- Boss alienígena que aparece al alcanzar cierto puntaje
- Sistema de colisiones, vidas y puntuación
- Empaquetado en Docker

Ejecución

```
1 docker run -it \  
2     -e DISPLAY=$DISPLAY \  
3     -v /tmp/.X11-unix:/tmp/.X11-unix \  
4     --rm cristian039/galaxy-rush
```

4 ROS: Comunicación entre Nodos

Tecnologías: ROS Noetic, Docker

Se implementó una arquitectura mínima tipo publicador-suscriptor donde un nodo emite mensajes y otro los escucha.

Ejecución en Docker

```
1 docker run -it --name tarea5 cristian039/ros-tarea5 ./start_talker.sh
2
3 docker exec -it tarea5 bash -c "source /opt/ros/noetic/setup.bash &&
  python3 listener.py"
```

Salida Esperada

```
1 [INFO]: Listener recibí : Hola desde ROS en Docker
```

5 Conclusión

Esta tarea permitió explorar la creación y ejecución de aplicaciones gráficas, juegos y sistemas de comunicación usando Docker. Integra conceptos clave del sistema operativo como gestión de procesos, visualización gráfica y ejecución en entornos aislados. Además, se fortalecieron habilidades de integración y despliegue multiplataforma.

Créditos

Cristian Olarte y Ana Vargas
Universidad Santo Tomas
Curso: Sistemas Operativos
Docente: [Diego Alejandro]