**FACULTAD DE INGENIERÍA**CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

**GUIA DOCUMENTADA**

INFORME PROYECTO

## Autores:

* Valdivia Seminario, Fernando José
* Cedric Pieer Pariona Miranda
* Pool Humberto Osores Monroy

Curso: Técnicas de Programación Orientada a Objetos

Docente del Curso: Lenin Froy Arce Huamán

Nombres:

Lima – Perú  
2025-2

# 1. Introducción

El presente trabajo busca aplicar los conceptos de Programación Orientada a Objetos (POO) en un contexto cercano a la vida cotidiana. Se eligió como escenario de observación un parque del barrio, donde se desarrollan diversas actividades que permiten identificar objetos, atributos, métodos y relaciones entre ellos.

# 2. Objetivos

***General:***

* Aplicar los principios de la POO para modelar un sistema sencillo basado en la observación de un parque.

***Específicos:***

* - Identificar objetos y actores presentes en el parque.
* - Modelar clases con sus atributos y métodos.
* - Representar las relaciones principales mediante un diagrama UML.

# 3. Metodología

Se realizó una observación directa del parque (o se modeló como caso de estudio) durante un periodo corto. Se registraron los actores presentes, sus acciones y las interacciones entre ellos. Posteriormente, se clasificaron los elementos observados en clases, atributos y métodos, y se representaron en un diagrama UML sencillo.

# 4. Desarrollo / Observaciones

Se identificaron los siguientes elementos y comportamientos en el parque:

* - Personas: niños jugando, adultos caminando, deportistas entrenando.
* - Juegos: columpios, resbaladeras, balancines.
* - Áreas verdes: césped, árboles, jardineras.
* - Trabajadores: personal de limpieza, jardinería y seguridad.

## 4.1 Identificación de Clases

A partir de las observaciones se modelaron las siguientes clases principales:

### Clase: Persona

* Atributos:

1. - nombre : string
2. - edad : int
3. - tipo : string (niño/adulto/deportista)

* Métodos:

1. - caminar()
2. - jugar()
3. - hacerEjercicio()

### Clase: Juego

* Atributos:

1. - tipo : string (columpio/resbaladera)
2. - estado : string (nuevo/en uso/dañado)

* Métodos:

1. - usar()
2. - reparar()

### Clase: ÁreaVerde

* Atributos:

1. - tipoPlanta : string
2. - tamaño : float (m2)
3. - estado : string (cuidado/descuidado)

* Métodos:

1. - regar()
2. - podar()

### Clase: Trabajador

* Atributos:

1. - nombre : string
2. - cargo : string (limpieza/seguridad/jardinero)

* Métodos:

1. - limpiar()
2. - vigilar()
3. - mantener()

## 4.2 Relaciones entre clases

Las relaciones principales identificadas son:

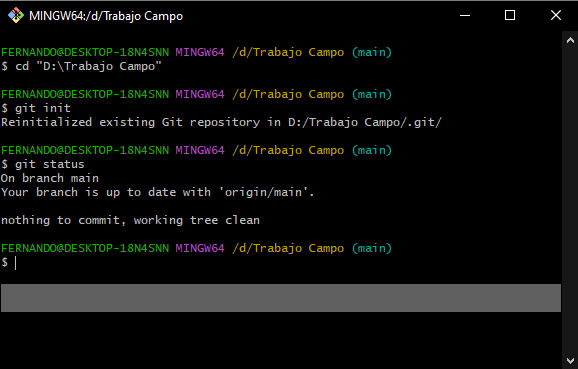
* - Una Persona puede usar un Juego.
* - Un Trabajador puede mantener tanto Juegos como ÁreasVerdes.

# 5. Diagrama UML (sencillo)

A continuación se presenta un diagrama UML básico que ilustra las clases y relaciones definidas:

# 6. Pantallazo GitHub

# 7. Pantallazo Consola



***Fin del documento.***