

# Documentación Tarea 1 - Dron y su control como objeto de software

Nicolás Aguayo, Alejandro Peralta, José Bórquez

18 de mayo de 2020

## 1. Introducción

Para la tarea se hace uso del lenguaje de programación **JAVA** que permite la utilización de objetos llamados clases y funciones asociadas llamadas métodos. Aunque al principio fue curioso la incursión a la programación a objetos (debido a la costumbre a trabajar en lenguajes como C y MatLab) se le encontró muchos beneficios al tipo de programación, como en el trato de excepciones y el manejo de las propiedades de los objetos.

## 2. Diagrama de Clases

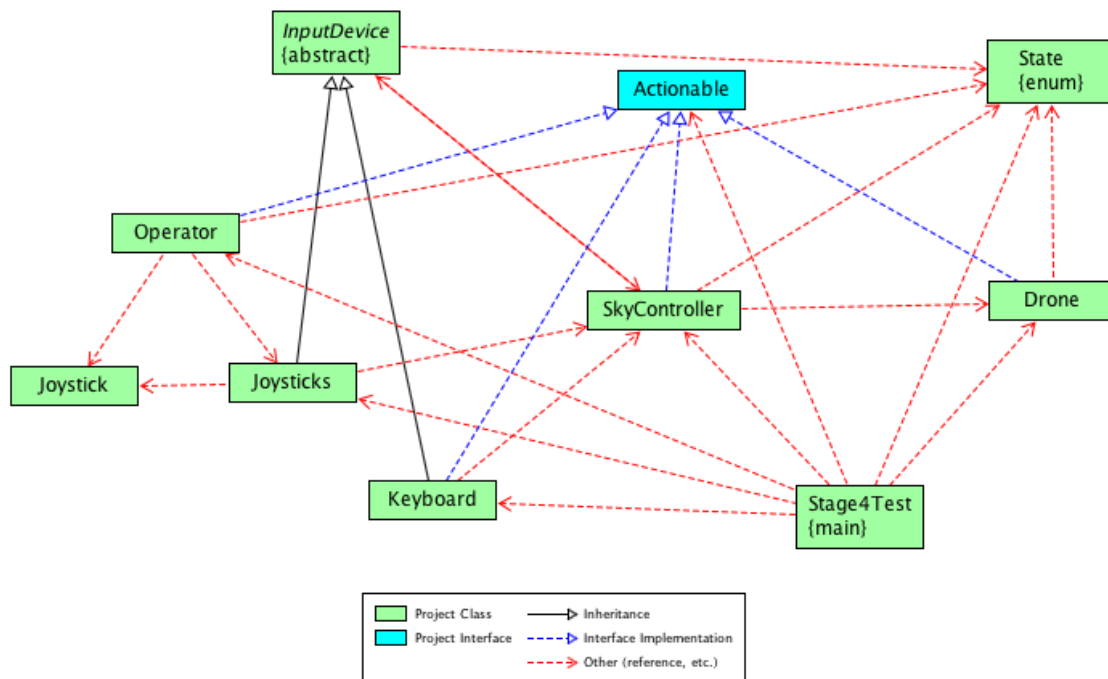


Figura 1: Diagrama de Clases de la Etapa 4

### 3. Explicación de la Solución

Para la realización de la tarea se hizo uso de cuatro etapas diferentes que tenían un objetivo individual, para luego, al asociarlo todo en la etapa 4 se lograran los objetivos propuestos de la tarea, siguiendo la metodología iterativa incremental.

**Actionable:** Interfaz que contiene el método `takeAction`.

**Drone:** clase del objeto dron, este contiene su posición, velocidad y archivo de escritura.

**DroneState:** clase *enum* que determina los estados del dron.

**InputDevice:** clase abstracta que configura las acciones que debe realizar un dispositivo de entrada.

**Joystick:** clase del mando análogo que se controla.

**Joysticks:** clase tipo de `InputDevice`, esta utiliza archivos de entrada.

**Keyboard:** clase tipo de `InputDevice`, esta utiliza el teclado.

**Operator:** clase que simula las acciones de un operador real, mientras obtiene la información de un archivo de entrada.

**SkyController:** clase del controlador del dron, este le envía la información que ejecuta el operador.

**StageXTest:** clase main de cada etapa.

En el desarrollo final el dron1 logra ser comandado por un operador que lee un archivo de entrada .csv que contiene las instrucciones para dibujar la palabra USM, a su vez otra instancia de Dron (dron2) es manejado vía teclado en tiempo de ejecución del programa. Ambos entregan un archivo de salida en formato .csv, se utilizó el programa Matlab para dibujar las trayectorias seguidas de ambos drones, ver figura 2.

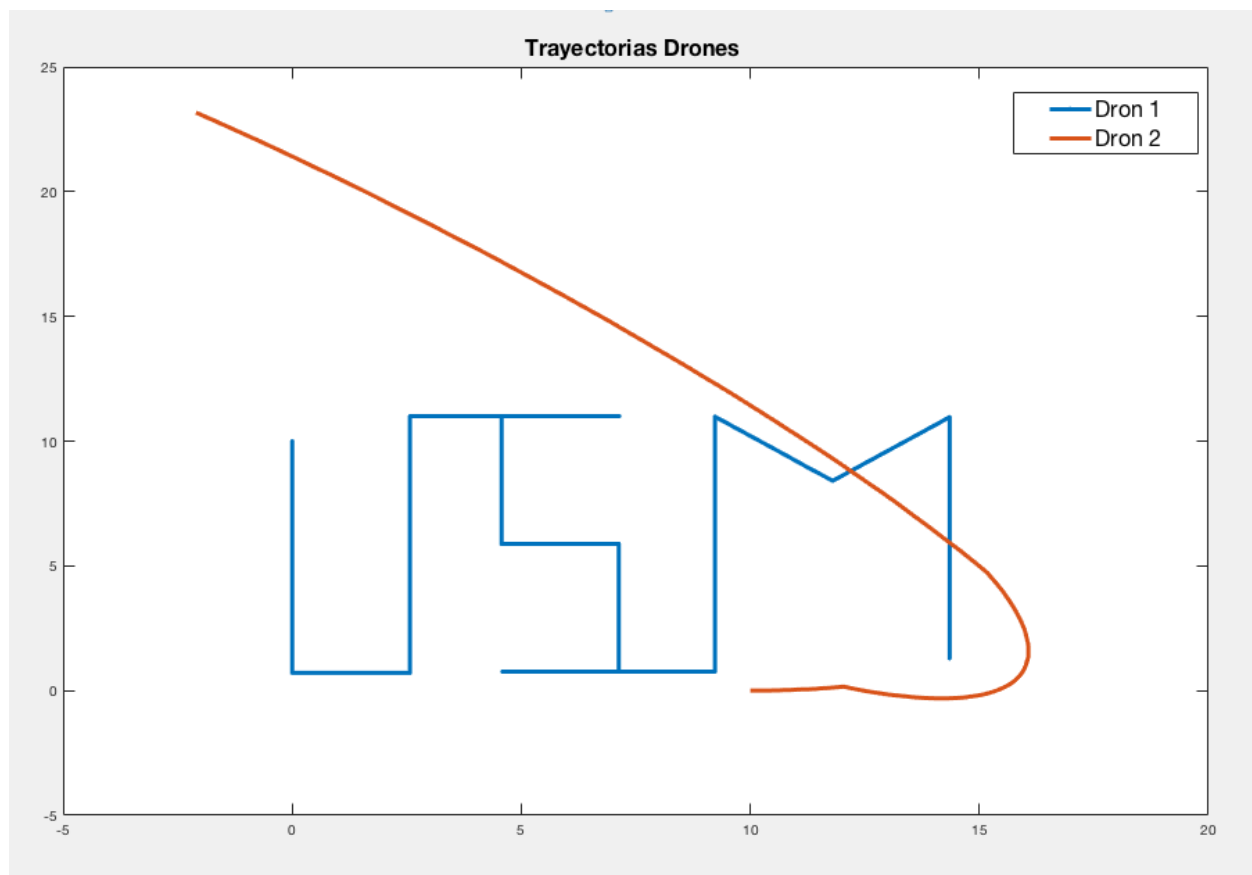


Figura 2: Trayectoria de Drones, Dron1 según entrada Joysticks, Dron2 según entrada Keyboard.

## 4. Dificultades y Soluciones

### 1. Formateo de variables de salida.

Al momento de escribir un flotante por pantalla surgió la dificultad de imprimirlo con punto en vez de coma, esto generaba problemas para leer el archivo csv. Fue solucionado cambiando el formato a US en la clase Locale.

### 2. Uso de clase abstracta `InputDevice()`.

Una de las dificultades experimentadas fue la utilización de la clase abstracta `InputDevice` propuesta en los códigos de ayuda, en un comienzo parecía agregar complejidad extra al código de manera innecesaria, sin embargo, una vez avanzadas las etapas se convirtió de mucha utilidad para cambiar el dispositivo que maneja el dron.

### 3. Identificación de partida.

Entender bien donde se tenía que partir para poder atacar la programación del código fue una de las más grandes dificultades, pues aunque se supiera que había que hacer y cómo hacerlo, el cómo enfrentar un problema necesito un poco más de tiempo.