

# ESCUELA COLOMBIANA DE INGENIERIA

## PROGRAMACIÓN ORIENTADA A OBJETOS

### PROYECTO INICIAL Ciclo No. 4 2022-2

#### REFACTORING Y EXTENSIÓN

El proyecto inicial tiene como propósito desarrollar una aplicación que permita simular una situación inspirada en el Problem B de la maratón de programación internacional 2020 **The Cost Speed Limits**

#### CUARTO CICLO

El objetivo de este ciclo es perfeccionar y extender el simulador cumpliendo los criterios de calidad de funcionalidad y extensibilidad.

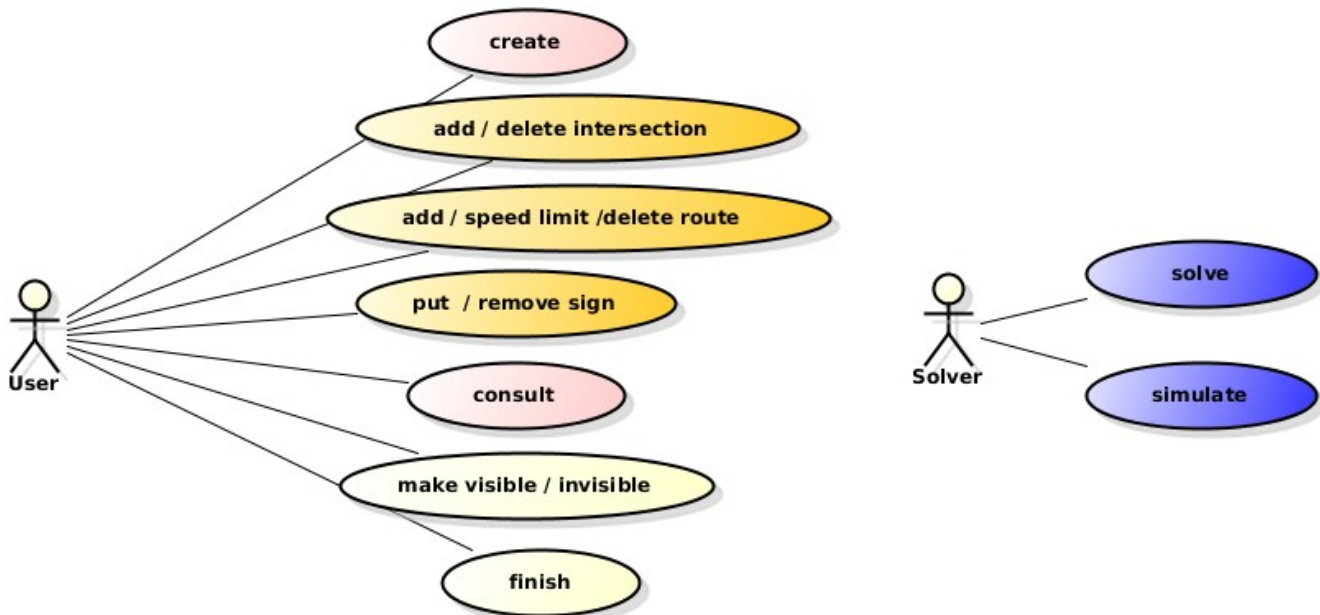
Para esto se espera que en esta entrega se tenga:

1. Estructura del sistema en dos paquetes: `shapes` y `icpc`
2. Refactorización del paquete `shapes` aprovechando el mecanismo de herencia. El paquete debe figurar completo.
3. Refactorización y extensión del paquete `icpc` para incluir los nuevos requisitos

#### NUEVOS REQUISITOS FUNCIONALES

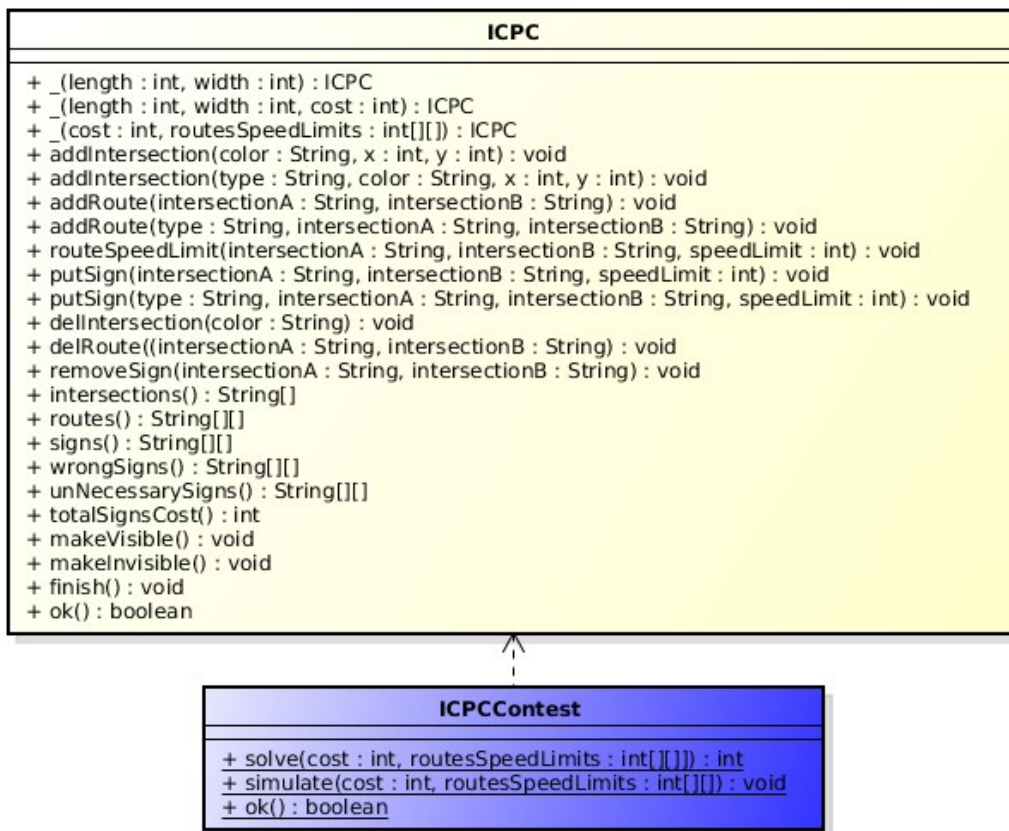
El simulador debe poder manejar diferentes tipos de intersecciones: **normal** (la que tenemos), **hermit** (solo permite una ruta), **needy** (si le quitan todas sus rutas, desaparece) ; diferentes tipos de rutas: **normal** (el que tenemos), **rebel** (no deja que le coloquen señales), **fixed** (no permite que la quiten) y diferentes tipos de señales: **normal** (el que tenemos), **twin** (siempre va con una gemela que se ubica en el otro extremo de la ruta), **cautious** (ajusta su límite al mínimo de las rutas de la ciudad en la que se ubica).

**Ustedes deben proponer un nuevo tipo de alguno de los elementos anteriores e incluirlo en ipc.**



Extensión: **intersection, route, sign**

## REQUISITOS DE DISEÑO Y CONSTRUCCIÓN



**El creador masivo y ICPCContest sólo usan elementos normales**

## REQUISITOS DE USABILIDAD

Los elementos de diferentes tipos debería poder distinguirse claramente.

## REQUISITOS DE ENTREGABLES

Los productos esperados para esta entrega son:

1. Diseño completo en la herramienta astah.  
Adicionar el diagrama de paquetes.  
**En astah, crear un diagrama de clases (cambiar el nombre por Package Diagram0 )**
2. Código siguiendo los estándares de documentación de java.
3. Código de pruebas de unidad que cubran los métodos desarrollados: `ICPCC4test`.  
Las pruebas de `ICPCCC4test` será una creación colectiva siguiendo el protocolo definido.
4. Código de dos pruebas de aceptación que evidencien lo mejor del proyecto: `ICPCAtest`  
Las pruebas deben incluir las esperas necesarias y la pregunta al usuario sobre si la acepta.
5. Documento de retrospectiva. (7 preguntas ver ciclo uno)  
Es necesario incluir la retrospectiva de este ciclo y de los anteriores.

## REQUISITOS DE ENTREGA

Los productos los deben publicar en el espacio preparado en moodle en un archivo .zip con un nombre igual a la concatenación de los apellidos de los autores, ordenados alfabéticamente.

Publicar productos : Semana 10 Sábado 22 Octubre