

ESCUELA COLOMBIANA DE INGENIERIA

PROGRAMACIÓN ORIENTADA A OBJETOS

PROYECTO INICIAL Ciclo No. 4 2015-2

REFACTORING Y EXTENSIÓN

PROYECTO INICIAL

El proyecto inicial tiene como propósito desarrollar una aplicación que permita simular la situación planteada en el **Problema E** de la maratón de programación internacional 2015 *Evolution in Parallel*

TERCER CICLO

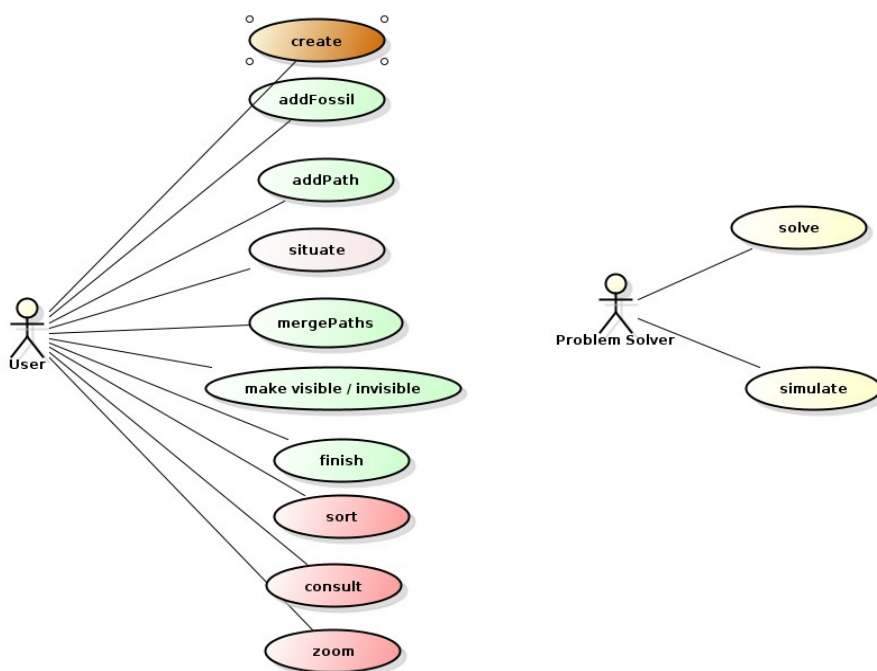
El objetivo de este ciclo es perfeccionar y extender el simulador y el solucionador de *Evolution in Parallel* para garantizar la calidad del mismo considerando los criterios de funcionalidad y extensibilidad.

Para esto se espera que en esta entrega se tenga:

1. Estructura del sistema en tres paquetes: `shapes`, `evolution` y `evolutionContest`.
2. Refactorización del paquete `shapes` aprovechando el mecanismo de herencia. El paquete debe figurar completo.
3. Refactorización y extensión del paquete `evolution` para incluir diferentes tipos de células.

REQUISITOS FUNCIONALES

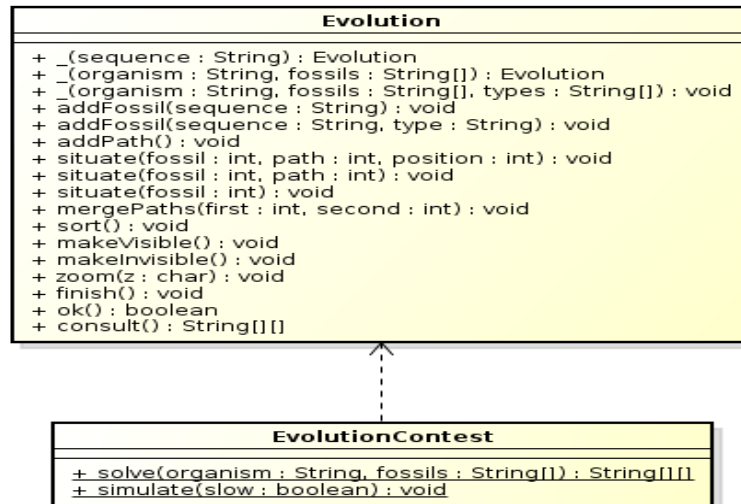
En el simulador debe manejar diferente tipos de células: **normal** (la que tenemos en el momento), **top** (únicamente se ubica como última de la línea de evolución y no permite que otra célula se ubique encima), **magical** (se transforma en la célula necesaria para ocupar la posición indicada, si es posible), **destroyer** (se ubica en el path que le piden eliminando las células que le impiden ubicarse allí), y una definida por ustedes.



REQUISITOS DE INTERFAZ

Los diferentes tipos de células deben lucir diferentes

REQUISITOS DE DISEÑO



OTROS REQUISITOS

Para esto se espera que en esta entrega se tenga:

1. Diseño completo en la herramienta astah
Diagrama de paquetes con relaciones
Diagrama de clases con atributos y métodos privados y públicos.
(No incluya atributos y métodos de las clases correspondientes a shapes)
Diagrama de secuencia completo de cada uno de los métodos de las clases *Evolution* y *EvolutionContest*
2. Código siguiendo los estándares de programación java.
No olviden que el código de los métodos no debe ocupar más de una pantalla.
3. Código de pruebas de unidad que cubran los métodos de las clases . Las pruebas se deben preparar en modo invisible.
Las clases *EvolutionTest* y *EvolutionContestTest* será una creación colectiva usando el wikis correspondientes.
Los nombres de los casos de prueba deberán incluir la identificación de los autores. Por ejemplo, **segunDA**deberia (DA: Iniciales de los primeros apellidos en orden alfabético).
4. Código de pruebas de aceptación de casos interesantes. Las pruebas se deben preparar en modo visible.
5. Documento de retrospectiva.

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. El .zip debe contener el diseño (.astah), la construcción (*.java) y la **retrospectiva (.doc)**.

Publicar productos a revisión de pares	: Miércoles 21 de octubre
Presentar revisión de pares	: Jueves 22 de octubre
Publicar productos versión final	: Lunes 26 de octubre(antes de las 8:00 a.m.)
Presentación final	: Lunes 26 de octubre