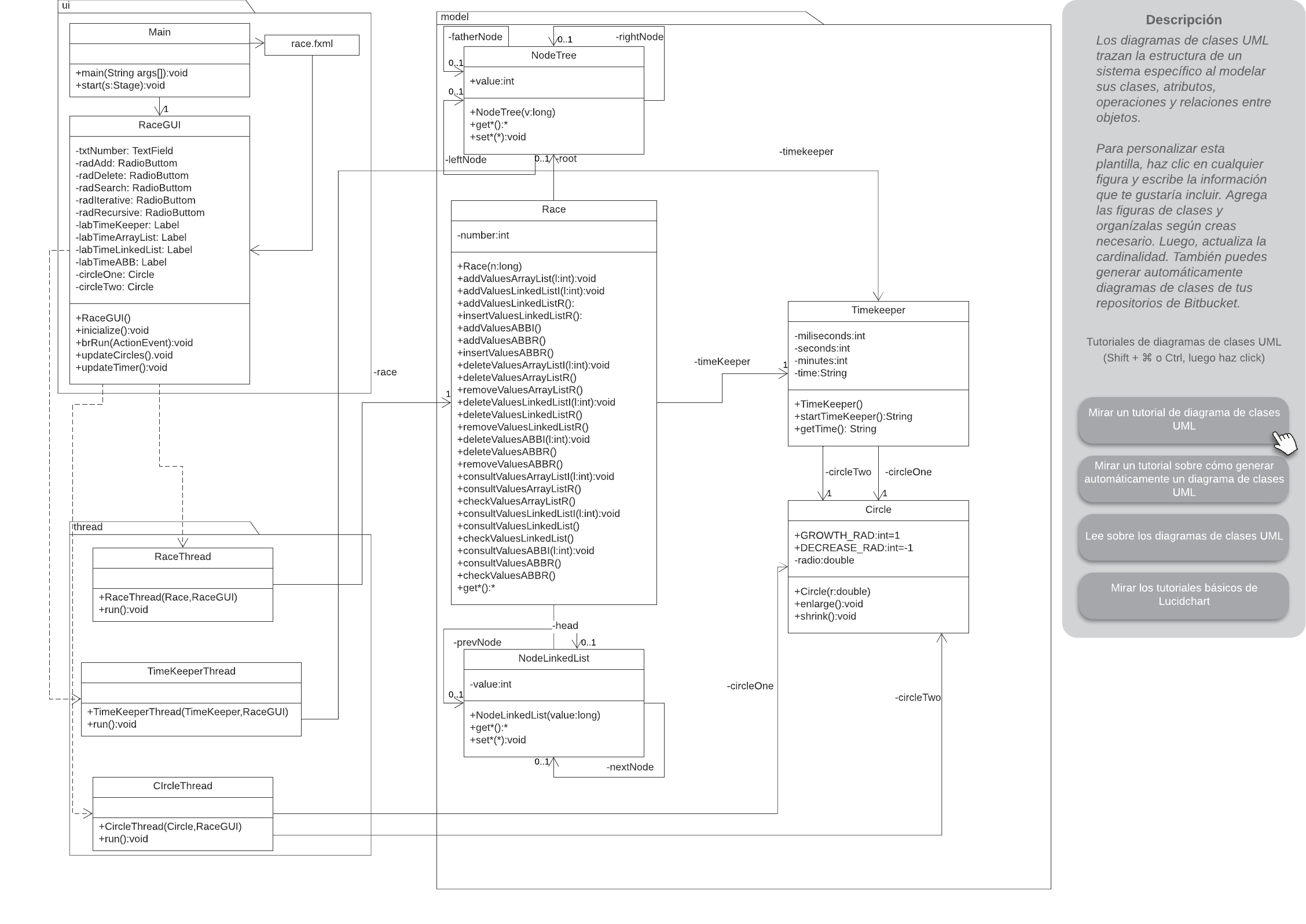
Laboratorio 3 Carrera de algoritmos

**Requerimientos funcionales**

El programa esta en la capacidad de:

* Realizar una carrera con las tres estructuras de datos: ArrayList, listas Enlazadas y Arboles Binarios, en la competencia de **añadir** el usuario debe digitar un numero N en el cual será el número de elementos que se debe añadir en cada estructura de datos, los elementos son generados aleatoriamente por el programa, consecutivamente el usuario debe digitar la modalidad de la carrera ya sea iterativa o recursiva. Además, se obtiene el tiempo que duro cada estructura de datos en añadir los elementos para luego mostrarlos en la pantalla para así definir un ganador.
* Hacer una carrera con las tres estructuras de datos: ArrayList, Listas Enlazadas y Arboles Binarios, en la competencia de **consultar** el usuario debe escribir un numero N en el cual será el numero de elementos que se debe consultar en cada estructura de datos, estos elementos son generados aleatoriamente, las estructuras de datos ya tienen N elementos aleatorios en el cual es la base donde se debe buscar los elementos, posteriormente el usuario selecciona la modalidad que desee hacer la carrera en la que puede ser iterativa o recursiva. Por otro lado, se toma el tiempo que ha durado cada estructura en hacer esa consulta para si mostrarlo en el sistema y poder ver quien gano la carrera.
* Elaborar una carrera con las tres estructuras de datos: ArrayList, Listas Enlazadas y Arboles binarios, en la competencia de **eliminar** el usuario debe anotar un numero N en el cual será el numero de elementos que se debe eliminar de cada estructura de datos, dichos elementos a eliminar son aleatorios, cada estructura de datos contiene N elementos aleatorios, luego el usurario debe seleccionar la modalidad de la carrera donde puede ser recursiva o iterativa. Para terminar el sistema guarda el tiempo que duro cada estructura en eliminar los elementos y lo muestra en la pantalla, con el fin de saber que estructura gano la carrera.
* Mostrar un cronometro que se iniciara desde el momento que empieza la carrera y termina cuando la última estructura de datos termine su ejecución. Además, este cronometro cuenta con dos círculos que lo que hacen es agrandarse y achiquitarse generando un tipo de animación, la idea es que cuando uno de los círculos se agrande uno se achiquite, con el fin de mostrar que el cronometro está corriendo. Cuando el cronometro se detiene estos círculos dejan de hacer su animación.

**Diagrama de clases**

****