



# Documentación Tarea 3:

## ELO 329

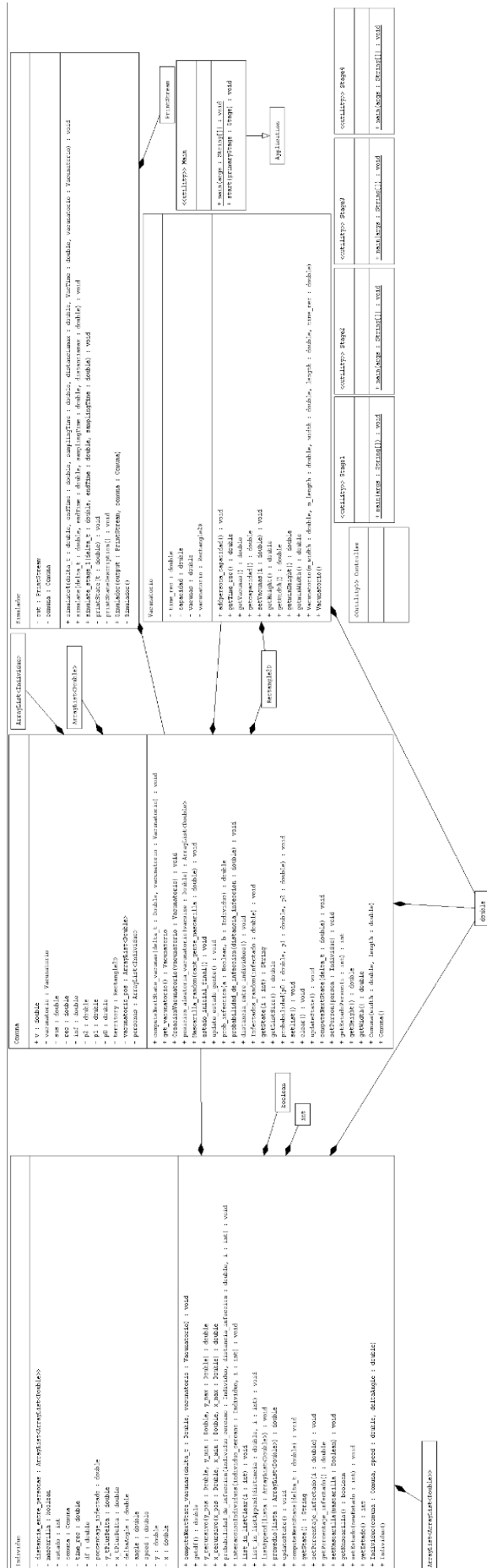
### “Simulador Gráfico en C++ de la Evolución de una Pandemia”

#### Integrantes:

- Catalina Latorre
- Eduardo Maldonado
- Eduardo Novoa
- Abraham Sánchez



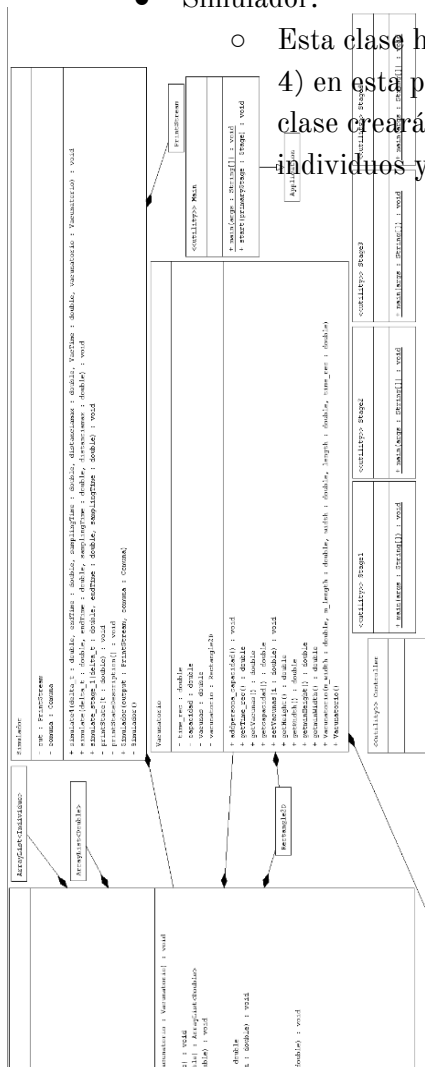
## Diagrama UML



## Explicación

Respecto al diagrama UML las clases que representan un mayor peso al programa son Individuo, Vacunatorio y Simulador:

- Pedestrian:
  - Esta clase hace referencia a una persona común y corriente. Como dice su nombre transeúnte. Dentro del programa esta puede estar sana, infectada, vacunada o recuperada. Los estados antes mencionados se les otorga a cada persona al ejecutar el programa y pueden ser modificados mientras se mueven por la “Comuna” (rectángulo de medidas entregadas). También cada persona posee una velocidad, un ángulo que lo direccionará en su movimiento y también puede usar, o no, mascarilla. Por último una persona vacunada o recuperada no puede volver a infectarse y en caso de estar sana la probabilidad de infectarse va ligada a la distancia a la que se encuentre de un individuo infectado, cuánto tiempo esté junto a él y si uno o ambos usan o no mascarilla.
- Vacunatorio:
  - Esta clase hace referencia a los sectores en donde se pueden vacunar las personas. Esta clase genera un sector aleatorio dentro de la “Comuna” y presenta un área. Solo las personas sanas pueden interactuar con estas zonas creadas, no pueden salir de esta zona por un tiempo específico y después saldrán como “persona vacunada”. Las personas infectadas no pueden interactuar con los vacunatorios.
- Simulador:
  - Esta clase hace referencia a la “ejecución” propia de las etapas (Stage 1, 2, 3 y 4) en esta participan las clases antes mencionadas dentro de la “Comuna”. Esta clase creará la instancia de una “Comuna” y a la vez la creación de los individuos y vacunatorios que la complementarán.





## **Dificultades en el desarrollo de la tarea y sus respectivas soluciones**

Para esta parte de la tarea, nos enfrentamos a varios problemas, los cuales más caben destacar, son los siguientes:

1. Funciones estáticas en c
  - Nos encontramos problemas con las funciones estadísticas, debido que a la hora de empezar a programar no teníamos mucho conocimiento acerca de cómo usar estas funciones, por eso tuvimos que investigar y experimentar de forma autodidacta.
2. Correr el código con más de un pc
  - Ya que estábamos trabajando en un pc principalmente a la hora de traducir nuestro código principal de Java a C++, tuvimos problemas a la hora de correr el código desde otros computadores, probablemente por el uso de distintos sistemas operativos, así que tuvimos que usar de manera casi obligatoria las máquinas virtuales, aunque después tuvimos problemas de memoria con estas mismas.
3. El uso qt y ejecutar nuestro código
  - Debido a nuestro desconocimiento a la hora de usar este framework, tuvimos que investigar y seguir tutoriales (como el provisionado por aula) para su instalación y uso. Pero nos vimos complicados a la hora de relacionar las funciones de C++ dentro de este framework, incluso como mencionamos anteriormente hubo un momento en donde el qt de un compañero mostraba un las posiciones respecto al tiempo del individuo y a otro solo le imprimían 0.