

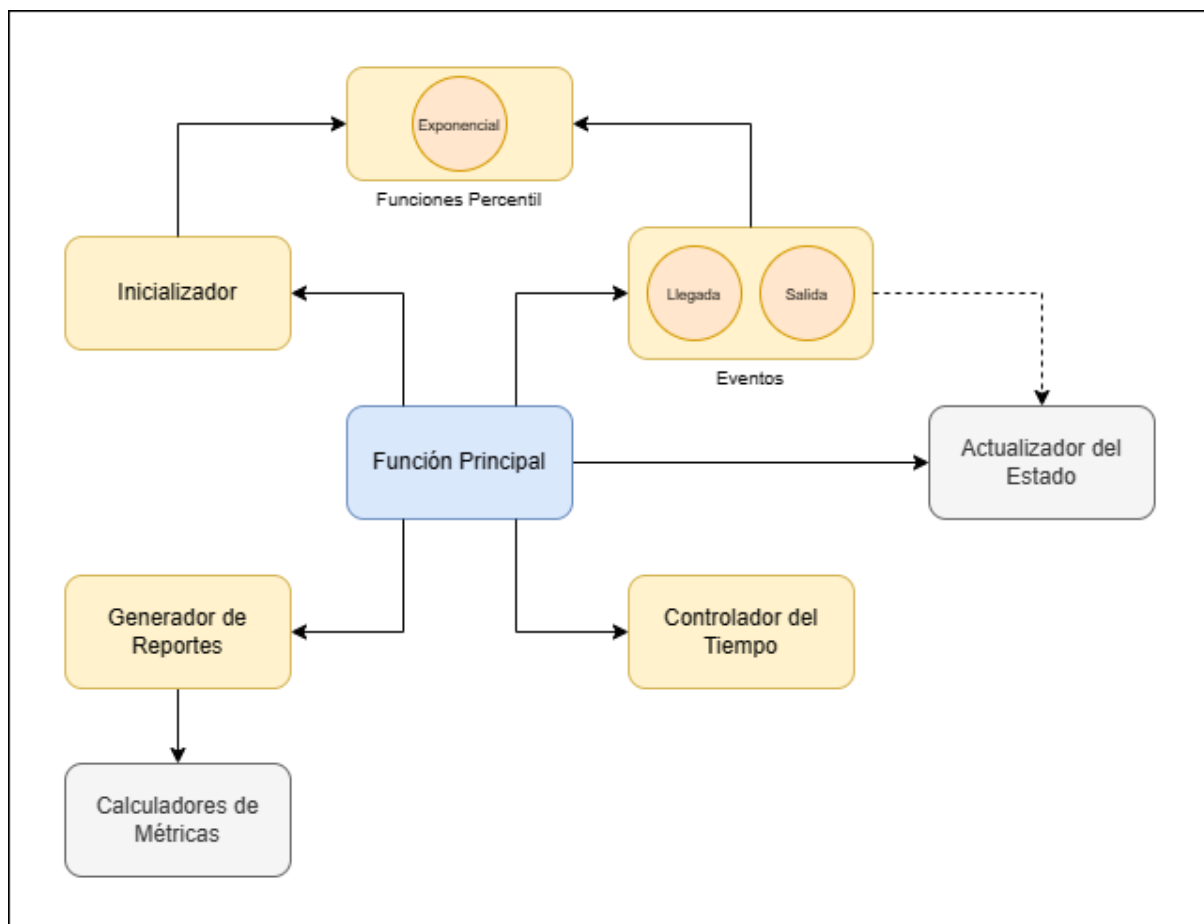
DISEÑO

ARQUITECTURA DEL SIMULADOR

Los componentes principales del simulador son los siguientes:

- Función principal
- Inicializador
- Funciones percentil: Exponencial
- Generador de reportes
- Controlador del tiempo
- Eventos: Llegada y Salida

Se incluye una función adicional de actualización del estado que podría considerarse como parte de cada uno de los eventos, sin embargo, en este diseño, el actualizador del estado es utilizado directamente por la función principal y no por cada función de eventos. Esto debido a que la implementación del actualizador es independiente del tipo de evento, y por eso se dejó de esta manera como si fuera un séptimo componente en la arquitectura.

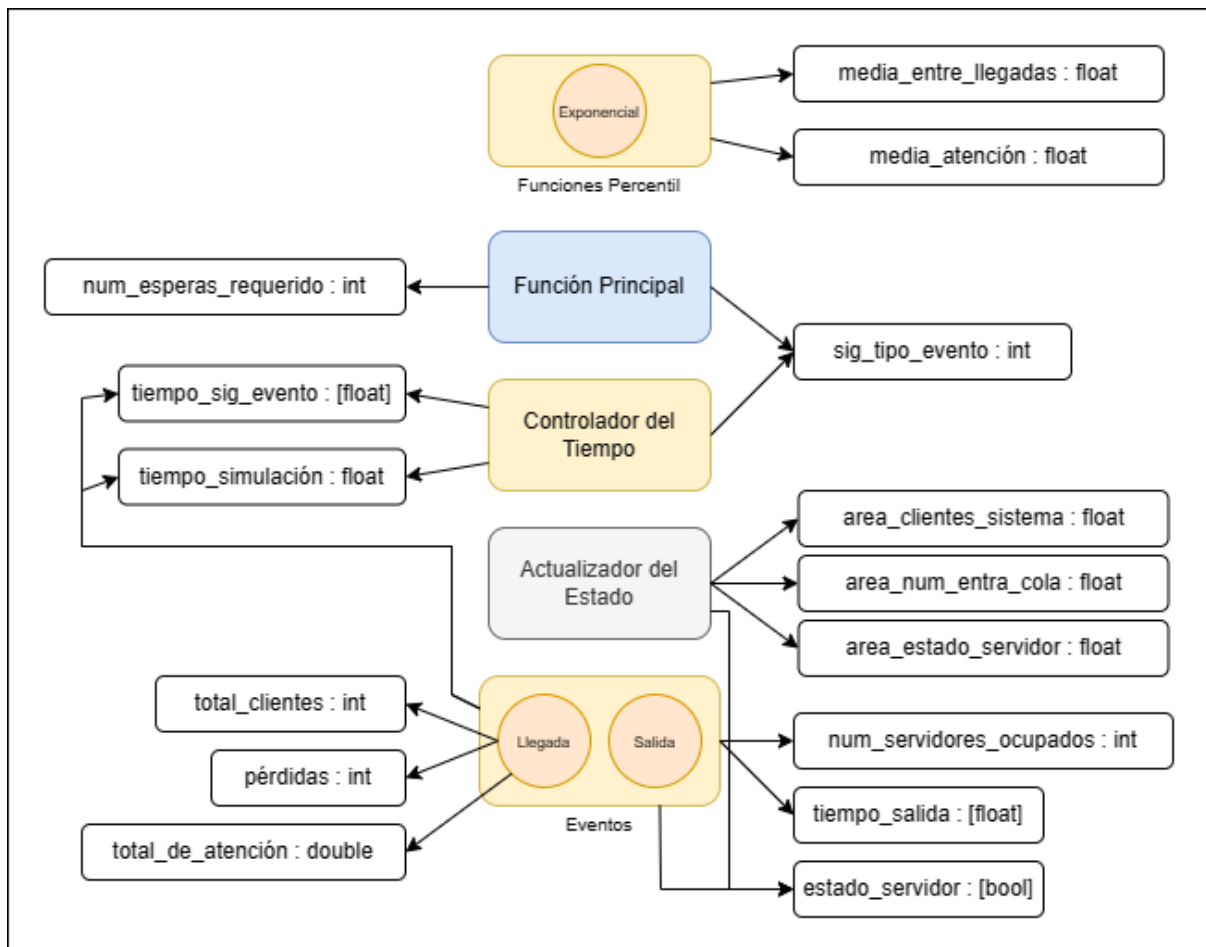


El diagrama presentado indica lo siguiente:

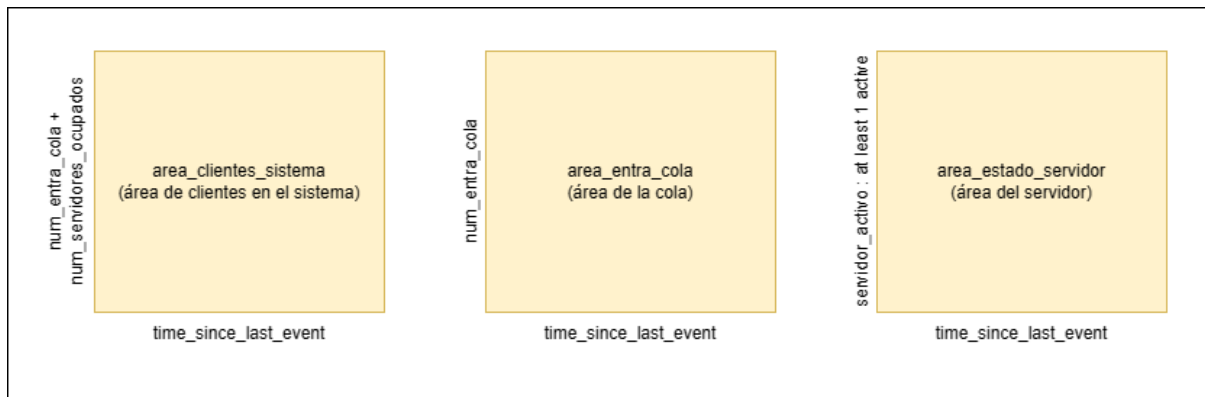
- La función principal utiliza el inicializador, las funciones de eventos, el generador de reportes, el controlador del tiempo y el actualizador del estado.
- El inicializador y las funciones de eventos utilizan las funciones percentiles, que en este caso solo es una, y es la exponencial
- El generador de reporte utiliza las funciones calculadoras de métricas

VARIABLES DEL SIMULADOR

En el siguiente diagrama se muestran las variables principales del simulador y qué componentes utilizan. El inicializador y el generador de reportes utilizan casi todas estas variables y por eso no se incluyen en este diagrama.



En el siguiente diagrama se muestra cómo se muestra la representación gráfica de las áreas que se van acumulando en cada actualización del estado del simulador.

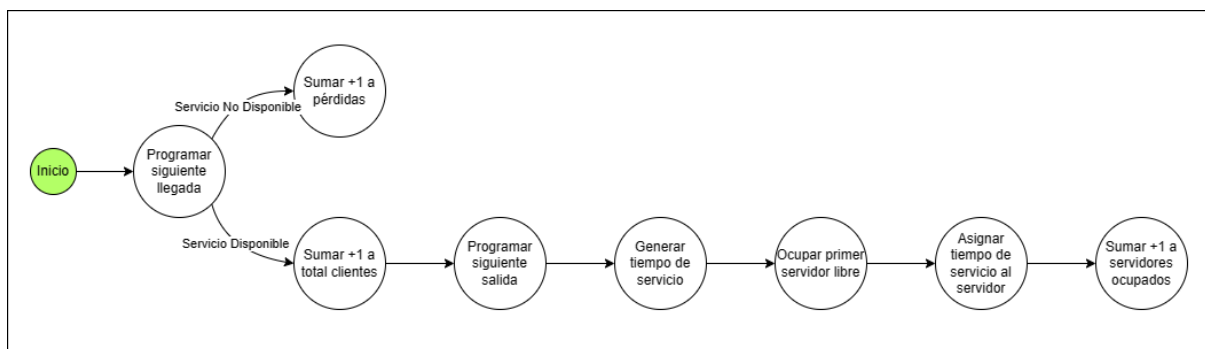


El estado del servicio (es decir de los servidores como un todo) se decide de la siguiente manera: El servicio está disponible si al menos un servidor está libre. El servicio no está disponible si todos los servidores están ocupados.



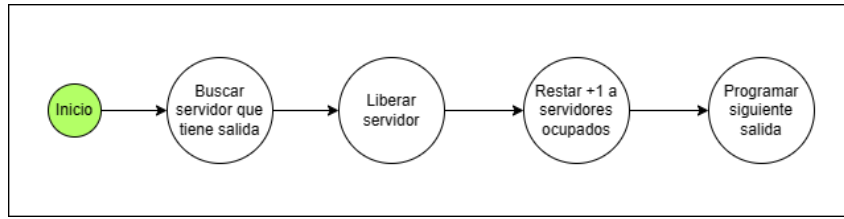
ALGORITMOS DE LOS EVENTOS: LLEGADA Y SALIDA

El algoritmo de llegada se presenta en el siguiente diagrama:



Se inicia programando la siguiente llegada. Luego se verifica el estado del servicio. Si no está disponible, se genera una pérdida. Si está disponible, se incrementa el número de clientes del sistema, se programa la siguiente salida. Se define el tiempo de atención que tendrá el servidor. Se busca el primer servidor que esté libre y se cambia su estado a ocupado, definiendo su tiempo de atención. Al final se incrementa el número de servidores ocupados en 1.

El algoritmo de salida se presenta en el siguiente diagrama:



Se inicia buscando el servidor que tiene salida comparando los tiempos de salida que tienen asignados los servidores. Se libera el servidor. Se disminuye la cuenta de servidores ocupados en 1 y se programa la siguiente salida.