

# Proyecto: primera entrega

Pontificia Universidad Javeriana

Introducción distribuidos

Docente: Mariela Curiel Huerfano



Juan Camilo Merchán Loza

Jose Fernando Zuluaga Ortiz

Nicolas Daniel Vargas Ortiz

30 de marzo 2022

Bogotá D.C

## **1. Diseño de Sistema**

### **1.1. Diagrama de clases**

*Anexo: Diagrama de clase.pdf*

### **1.2. Diagrama de secuencias**

*Anexo: Diagrama de secuencia.pdf*

## **2. Modelos**

### **2.1. Interacción**

- Asíncrono: ya que no es requerido la sincronización de los relojes donde se ejecuten los procesos.

### **2.2. Fallos**

- Fail-stop: constantemente se está mandando request a los monitores, si estos no responden con un replay entonces se asume que el proceso murió y se lanza un nuevo monitor.
- Ruptura: un sensor puede dejar de funcionar. Ya que ningún proceso está verificándolos pasaría inadvertido el fallo hasta que un usuario se percate de esto. Sin embargo, no se vería afectado críticamente el funcionamiento del sistema ni realizaría acciones incorrectas.
- Omisión: si existe una falla en la comunicación se perderían los mensajes.
- Bizantino: se puede generar un fallo arbitrario a nivel de canales donde el contenido se vea comprometido de manera que este se corrompa.

### **2.3. Seguridad**

- Puesto que no se tiene una validación del origen y el destino del mensaje esta puede ser interceptada por un tercero
- Debido a que la información que enviamos no tiene ningún tipo de encriptación esta puede ser modificada o copiada por un tercero.

### 3. Protocolo de Pruebas

Nuestro protocolo de pruebas consta de dos escenarios:

- **Escenario 1:**

Constaremos de la ejecución de 6 sensores, 2 sensores para cada tipo, estos sensores se iniciarán con la configuración correspondiente al archivo de cada tipo, y tres monitores, los cuales reciben información de dos sensores, correspondientes al tipo de información que registra, pH, oxígeno o temperatura.

- **Escenario 2:**

Como segundo escenario, para probar la capacidad de un solo monitor, se tendrá los seis sensores de solo un tipo de información. De esta forma, se puede ver el rendimiento de los sensores y del tiempo que el monitor tiene para gestionar y recibir la información de estos sensores.

### 4. Elementos que afectan el rendimiento

Para este caso evaluaremos el caso de aumentar la carga en nuestro sistema

- **Hipótesis**

Se llega a la hipótesis de que el aumento de sensores podría generar una congestión en el canal ocasionando un aumento en la latencia y por ende un aumento en el retraso de los mensajes y la pérdida de estos

- **Variables para medir**

Las variables que medir serán aquellas que traten el rendimiento de los sistemas, es decir, aquellos presentes en el modelo de iteración

- Latencia: la tardanza en la transmisión de paquetes en la red podría aumentar debido a que se aumentan los tiempos de espera para la transmisión de mensajes.
- Ancho de banda: Al aumentar el número de sensores se espera un aumento en la transmisión de mensajes y por ende un aumento en el uso del ancho de banda.
- Fluctuación: Debido al aumento de la congestión en los canales, prevemos una variación en la llegada de los mensajes y por ende una variación en el tiempo de ejecución de ciertos procesos.

- **Herramientas de medición**

- Jmeter: software open source hecho en java y utilizado para cargar test de funcionalidad y medir el rendimiento. JMeter mide la latencia justo antes de enviar la solicitud hasta justo después de recibir la primera respuesta.