



**PONTIFICIA UNIVERSIDAD JAVERIANA**  
**FACULTAD DE INGENIERIA**  
**DEPARTAMENTO INGENIERIA DE SISTEMAS**  
**Proyecto Introducción a Sistemas Distribuidos**  
**Período Académico 2026-10**

## Gestión Inteligente de Tráfico Urbano

40%

### Objetivos

- Desarrollar una solución a un problema de estructura distribuida.
- Utilizar patrones de comunicación síncronos y asíncronos.
- Resolver problemas que se presentan en sistemas distribuidos, tales como fallas en los componentes y persistencia de datos.
- Reconocer atributos de calidad (ej. desempeño, resiliencia) asociados a la implementación de un sistema distribuido

### Descripción del Sistema a Desarrollar

El objetivo del proyecto es diseñar e implementar una plataforma distribuida para la gestión inteligente del tráfico urbano, cuyo propósito es monitorear, analizar y reaccionar ante condiciones de tráfico utilizando múltiples componentes distribuidos que se comunican mediante la librería ZeroMQ (ZMQ). <https://zeromq.org/>

El sistema simula una ciudad con intersecciones controladas por semáforos inteligentes y una red de sensores de tráfico (espiras inductivas, cámaras de tráfico y sensores GPS simulados) que generan eventos en tiempo real, relacionados con volumen vehicular, velocidad promedio y nivel de ocupación de las vías. La ciudad se representa con una matriz o cuadrícula de  $N \times M$ , donde  $N$  es la fila representada por una letra y  $M$  es la columna representada por un número, y usaremos la notación  $INT\_CK$ , para representar la intersección de la fila  $C$  con la columna  $K$ . Por ejemplo,  $INT\_C5$ , representa la intersección de la fila  $C$  columna 5. Los sensores están ubicados en varias de las intersecciones.

Los eventos generados son procesados por un servicio de analítica con el propósito de:

- Recopilar y almacenar información de tráfico.
- Analizar condiciones de congestión vehicular.
- Tomar decisiones de control sobre semáforos.
- Consultar situaciones de tráfico en un determinado momento

- Emitir acciones de cambio de luz en los semáforos en momentos particulares

## FUNCIONAMIENTO DEL SISTEMA Y SUPUESTOS

Los sensores, eventos generados y servicios se ubicarán en tres máquinas (PC1, PC2 y PC3), tal y como se describe a continuación, y se muestra en la figura 1.

Vamos a suponer que:

- Todas las vías van en un único sentido
- El cambio de semáforo será únicamente de **verde a rojo y de rojo a verde**, no se utiliza el amarillo

### Componentes PC 1

Los sensores de tráfico generan eventos periódicos y los envían de forma asíncrona. Los sensores de tráfico se implementan como procesos simulados que generan eventos a partir de variables aleatorias. Dichos eventos se envían al broker ZeroMQ mediante el patrón PUB/SUB, garantizando comunicación asíncrona y desacoplada incluso cuando los componentes se ejecutan en el mismo PC. El **broker ZeroMQ** actúa como un **intermediario de mensajería** que **desacopla** a los productores y consumidores de datos dentro del sistema distribuido. El broker se suscribe a los tres tópicos asociados a los tres tipos de sensores, recibe los eventos generados por los sensores y los envía al nodo de procesamiento y control (PC2) en donde se almacenan y analizan.

Existen 3 tipos de sensores: Espiras inductivas, Cámaras de tráfico y GPS; cada uno se encarga de medir un evento en particular:

*Ejemplo de evento de un sensor tipo cámara: (EVENTO\_LONGITUD\_COLA (Lq))*

```
{
  "sensor_id": "CAM-C5",
  "tipo_sensor": "camara",
  "interseccion": "INT-C5",
  "volumen": 10,           //Número de vehículos en espera de cambio de semáforo
  "velocidad_promedio": 25, //Velocidad máxima 50 km/h
  "timestamp": "2026-02-09T15:10:00Z"
}
```

*Ejemplo de evento de un sensor tipo Espira inductiva (EVENTO\_CONTEO\_VEHICULAR (Cv))*

```
{
  "sensor_id": "ESP-C5",
  "tipo_sensor": "espira_inductiva",
  "interseccion": "INT-C5",
  "vehiculos_contados": 12, //Número de vehículos que han pasado sobre la espira
  "intervalo_segundos": 30, //cada 30 segundos, coincidiendo con los cambios de semáforo
  "timestamp_inicio": "2026-02-09T15:20:00Z",
  "timestamp_fin": "2026-02-09T15:20:30Z"
}
```

*Ejemplo de evento de un sensor tipo GPS (EVENTO\_DENSIDAD\_DE\_TRAFICO (Dt))*

```
{
  "sensor_id": "GPS-C5",
  "nivel_congestion": "ALTA",      //cambia el valor dependiendo de la velocidad promedio
  "velocidad_promedio": 18,      // ALTA (<10), NORMAL (entre 11 y 39), BAJA (> 40)
  "timestamp": "2026-02-09T15:20:10Z"
}
```

Los estudiantes deben determinar los datos de inicialización de estos procesos, por ejemplo: posición en la ciudad del sensor, cuadrículas que se abarcan, tiempo entre generación de un evento y otro, etc.

## Componentes PC 2

- El **servicio de analítica** se suscribe a los eventos generados por los sensores (PUB/SUB) vía el bróker ZMQ, recibe y procesa los datos para detectar congestión o anomalías a partir de reglas simples y envía la información directamente a la Base de Datos usando el patrón PUSH/PULL. Si se detecta una condición relevante, por ejemplo, dar prioridad al paso de una ambulancia, se generan eventos de control y toma decisiones, como extender fase verde de un semáforo. Para comunicar cambios en los semáforos, El servicio de analítica se comunica de forma asíncrona con el *servicio de control de semáforos*, enviando comandos de control que son ejecutados sin bloquear el procesamiento de otros eventos. Eventualmente el servicio de analítica puede recibir indicaciones directas del módulo de *Monitoreo y consulta* ubicado en el PC3, emitidas por un usuario, para obligar a un cambio de estado de los semáforos, independiente de los datos generados por los sensores. Este servicio imprimirá por pantalla mensajes que indiquen si, de acuerdo con reglas, el tráfico es normal, hay congestión, etc. y qué acciones se van a tomar, si es el caso.
- El **servicio de control** de semáforos ajusta el estado de los semáforos simulados cambiando de luz roja a verde y viceversa, de acuerdo con las órdenes recibidas del servicio de analítica. El servicio debe imprimir por pantalla las operaciones que va realizando.
- La base de datos réplica, localizada en el PC2, es actualizada constantemente de forma asíncrona, con el fin de servir de backup en caso de que ocurra un fallo en el PC3

## Componentes PC3

- El servicio de monitoreo y consulta permite a un usuario **consultar el estado del sistema o enviar indicaciones directas** al módulo de analítica. Con estas indicaciones directas se puede forzar el cambio de estado del sistema, por ejemplo, para priorizar el paso de una ambulancia. Se pueden hacer consultas históricas entre periodos de tiempo (p. ej. Horas pico) y consultas puntuales de información en una intersección en particular utilizando el patrón de comunicación REQ/REP. Como en el caso de todos los servicios anteriores, el servicio de monitoreo y consulta debe imprimir todas las operaciones que va realizando.
- **ARQUITECTURA GENERAL DEL SISTEMA**

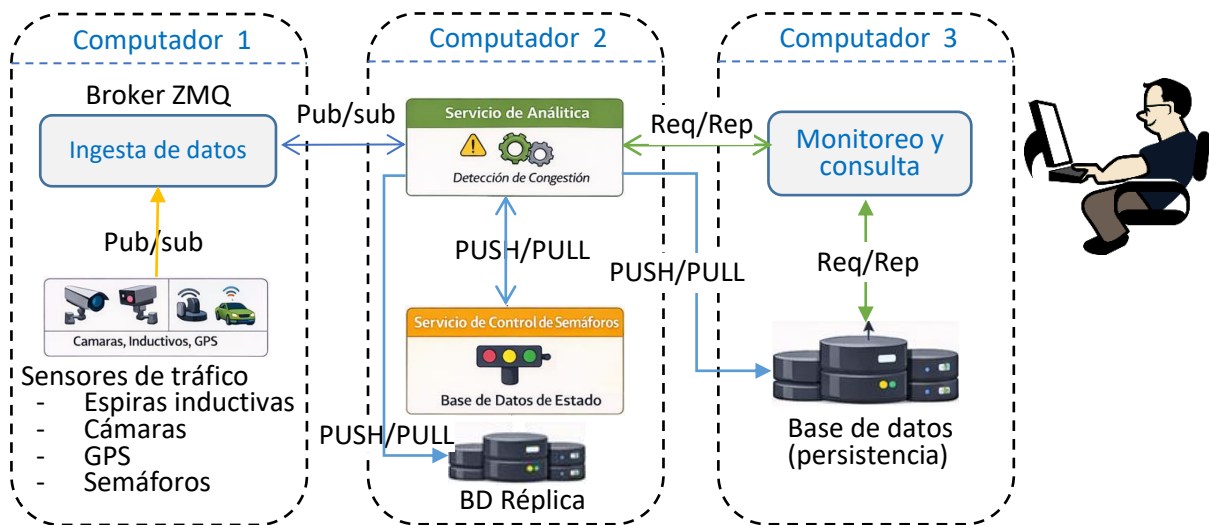


Figura 1: Componentes del Sistema.

Teniendo en cuenta las variables que miden los sensores:

- Espiras inductivas → conteo vehicular  $V$  (veh/min)
- Cámara → longitud de cola  $Q$  (No. Vehículos)
- GPS → densidad y velocidad promedio  $D$ ,  $V_p$  (veh/km, km/h)

Se deben definir rangos de condiciones normales de tráfico para establecer reglas simples, por ejemplo, para el **tráfico normal** se podría establecer la siguiente regla:

$$Q < 5 \quad \text{AND} \quad V_p > 35 \quad \text{AND} \quad D < 20$$

Cada grupo puede generar diferentes reglas, siempre y cuando cumplan con los parámetros establecidos y tengan un sentido lógico dentro del sistema.

El tiempo de espera para que la luz roja cambie a verde en condiciones normales es de 15 segundos.

Se deben establecer estados de circulación para:

1. Tráfico normal
2. Detección de congestión
3. Priorización de una vía (ola verde)

El servicio de analítica recibe la información y dependiendo de las reglas, realiza los cambios de luz verde o de la luz roja (suponemos que no hay transición con luz amarilla) o aumenta los tiempos de permanencia del semáforo en un determinado estado, dependiendo de la condición.

Los usuarios también pueden solicitar al servidor de consulta y monitoreo el cambio a luz verde de los semáforos de una vía, en casos especiales, por ejemplo, el paso de una ambulancia.

### Almacenamiento y Persistencia

La base de datos principal se encuentra en el PC3 y la réplica en el PC2; el servicio de analítica re-envía toda la información a las dos bases de datos utilizando un patrón de comunicación asíncrono, con el fin de mantener la persistencia y permitir consultas de estado de la red de semaforización.

**Procesos y número de computadores para la realización del proyecto:** la implementación debe corresponder a la arquitectura planteada con los correspondientes patrones de comunicación entre componentes, es obligatorio usar la librería ZeroMQ para las comunicaciones entre los diferentes procesos del proyecto (<https://zguide.zeromq.org/>). Si va a cambiar alguno de los patrones de comunicación debe justificarlo en la primera entrega.

Cada grupo debe definir las reglas para los 3 estados (Tráfico normal, congestión, priorización) y sincronizar el sistema de semaforización, así como las consultas de los usuarios al servicio de monitoreo y las posibles operaciones para cambios de estado, con sus parámetros.

**Fallas:** En su implementación deben considerar una posible falla del PC3. Si esto ocurre, todos los procesos deben comenzar a usar inmediatamente la réplica de la base de datos que se encuentra en el PC2. La operación será transparente para el cliente; el sistema debe continuar operando de forma ininterrumpida.

**Necesario para la evaluación:** el día de la sustentación es importante que se pueda observar:

- Estado de la BD (original y réplica) y como van quedando a medida que se dan los cambios de estado.
- Operaciones que se van realizando en los diferentes servicios.
- Se debe poder consultar los estados de congestión históricos y las situaciones puntuales de priorización de semaforización

## Medidas de Rendimiento

Una vez implementado el proyecto, el equipo realizará pruebas, para comparar **el diseño o implementación inicial con un diseño modificado**. El diseño original es el que se ha descrito en este enunciado. El diseño modificado **va a utilizar hilos en el servicio BrokerZMQ**

	Diseño solicitado en el proyecto	Diseño multihilos en el Broker ZeroMQ
<b>Numero de sensores generando información de la ciudad y tiempo entre generación de mediciones (Variables independientes o factores del experimento)</b>	Variables a medir (variables dependientes): <ul style="list-style-type: none"> <li>- Cantidad de solicitudes almacenadas en la BD en un intervalo de tiempo de 2 minutos.</li> <li>- Tiempo desde que el usuario solicita una acción hasta que el semáforo cambia.</li> </ul>	Variables a medir (variables dependientes): <ul style="list-style-type: none"> <li>- Cantidad de solicitudes almacenadas en la BD en un intervalo de tiempo de 2 minutos.</li> <li>- Tiempo desde que el usuario solicita una acción hasta que el semáforo cambia.</li> </ul>
<b>1 sensor de cada tipo generando datos cada 10seg</b>		
<b>2 sensores de cada tipo generando datos cada 5 seg.</b>		

**Tabla 1: Escenarios de prueba y variables a medir.**

**Rellene las tablas y realice gráficos de las variables dependientes en función de los factores o variables independientes. Comente los resultados obtenidos.** Qué diseño es más escalable, justifique su respuesta en función de los resultados

### PRIMERA ENTREGA (15%)

La primera entrega se **realizará el lunes de la semana 10**, para sustentar el martes en el horario de la clase.

La primera entrega consta de un informe donde se debe especificar:

- Modelos del sistema (arquitectónico, interacción, fallos y seguridad). Cómo se aplican los conceptos de estos modelos al proyecto.
- Diseño de **TODO** el sistema: Diagrama de despliegue, Diagrama de componentes, Diagrama de clases y Diagrama de Secuencia. Este diseño debe incluir el o los componentes para enmascarar las fallas del sistema.
- En el informe debe explicar: a) Como los procesos obtendrán la definición inicial de los recursos: número y tipo de sensores, tamaño de la matriz, número de semáforos, etc. b) Reglas c) Tipos de consulta que harán los usuarios. D) Ejemplos de indicaciones directas que le hará el servicio de Monitoreo al servicio de analítica.
- El protocolo de pruebas que utilizará para la entrega final (considere **todos** los tipos de prueba que deben realizarse a un sistema), haciendo énfasis en las pruebas de desempeño.
- Cómo va a obtener las métricas de desempeño de la tabla 1
- Adicionalmente deben tener implementado: Servicios del PC1 y PC2 y actualizaciones a la BD principal en el PC 3
- Código fuente de las funcionalidades implementadas.

El día de la sustentación, cada equipo tendrá 15 minutos para mostrar sus resultados y responder las preguntas que tengan los profesores.

### SEGUNDA ENTREGA (15%)

La entrega se realizará la semana 17 para sustentar en el horario de la clase. El día de la sustentación los integrantes del equipo deben **mostrar la funcionalidad completa del proyecto**. Deben estar presentes todos los integrantes del grupo. La sustentación es presencial.

La entrega se compone de:

- En un archivo .zip código fuente de los programas que conforman el sistema y un archivo README donde indique cómo ejecutarlo.
- Se debe complementar la documentación de la primera entrega. Los archivos fuente deben estar documentados.
- Un video de máximo 10 minutos donde explica los siguientes aspectos de su proyecto:
  - a. Distribución de componentes en máquinas.

- b. Parámetros de todos los tipos de procesos.
  - c. Cómo se distribuye la cuadrícula de la ciudad entre los diferentes sensores.  
Cómo se asignan los semáforos.
  - d. Librerías y patrones usados
  - e. Tratamiento de la falla
- Un informe de máximo 5 páginas donde expliquen los experimentos realizados y resultados obtenidos. Aparte de las especificaciones que se encuentran en el informe, deben agregar: **especificaciones de hw y sw donde se realizaron las medidas** y herramientas de medición utilizadas. Debe mostrar en tablas y gráficos los resultados obtenidos, así como un análisis de dichos resultados.

#### Equipos de Trabajo.

El proyecto se realizará en grupos de trabajo a designar.

**No puede existir replicación de documentos ni de código fuente entre grupos, lo cual se consideraría plagio.**

M. Curiel/R. Paez/J. Corredor/O. De Castro/R. Gonzalez

### Calificación I Entrega

**Valor 15%.**

Indicador	Valoración en puntos /5pts	Excelente	Competente	Deficiente
Informe (presentación, completitud)	1	1	[0.75, 0.5)	<0.5
Diseño del Proyecto	1.5	[1.5,1]	(1, 0.25]	< 0.25
Protocolo de pruebas	0.5	0.5	0.25	< 0.25
Modelos del Sistema (modelos de fallas, interacción y seguridad)	0.25	0.25	0.15	<0.15
Obtención de las métricas de rendimiento	0.25	0.25	0.15	<0.15
Implementación Inicial	1.5	[1.5,1.0]	(1.0, 0.75]	< 0.75
Total	5ptos			

<b>Indicador</b>	<b>Excelente</b>	<b>Competente</b>	<b>Deficiente</b>
Informe	La presentación del informe es impecable, sin problemas de ortografía o redacción. El informe contiene todos los aspectos solicitados	El informe tiene fallas menores en la presentación, ortografía y/o redacción. El informe contiene todos los aspectos solicitados	El informe tiene fallas importantes en la presentación, ortografía y/o redacción. No contiene todos los aspectos solicitados.
Diseño del Proyecto	Se presentan todos los artefactos de diseño exigidos en el enunciado: diagrama de componentes, diagrama de clases, diagrama de secuencia y diagrama de despliegue. Se incluyen todos los componentes del sistema final, incluyendo tolerancia a fallas y persistencia. Los diagramas están correctos.	Los diagramas incompletos o algunos diagramas están incorrectos.	No se realizan los diagramas exigidos en el enunciado o los diagramas presentados están incorrectos. No se considera TODO el sistema que se va a implementar en el diseño.
Protocolo de pruebas	En el protocolo presentado por los estudiantes, las pruebas son suficientes para evaluar la funcionalidad de la aplicación con y sin la presencia de fallas.	En el protocolo presentado no se contemplan todas las pruebas importantes. Está incompleto.	Los estudiantes no presentan protocolo de pruebas funcionales.
Modelos de sistema	Se hace una descripción de los modelos fundamentales adaptados al proyecto.	Se describen parcialmente los modelos fundamentales adaptados al problema a resolver.	No se presenta ni se describen los modelos fundamentales o la descripción no está relacionada con el proyecto a resolver.



Obtención de las métricas de rendimiento	Se describe de forma clara y completa las herramientas y/o metodología para obtener el valor de las métricas.	En el informe no está suficientemente claro el procedimiento para obtener los valores de las métricas.	No se mencionan en el informe ni el procedimiento ni las herramientas para obtener el valor de las métricas de rendimiento.
Implementación Inicial	Todas las funcionalidades requeridas en el enunciado para la primera entrega están implementadas correctamente. El sistema funciona en más de un computador (físico o máquina virtual)	De los siguientes aspectos, solo 2 fueron implementados correctamente: <ul style="list-style-type: none"> <li>• Solicitud de operaciones de devolución y renovación <b>desde los PS hasta los Actores.</b></li> <li>• Mecanismo para generar los requerimientos (lectura de un archivo o usando un generador de carga)</li> <li>• Ejecución en 2 máquinas.</li> </ul>	La implementación inicial es bien deficiente, se observan falencias en 2 o mas de los de los puntos mencionados en la celda anterior

## Calificación II Entrega

**Valor 25%**

**Informe de rendimiento 10% (Evaluado sobre 5ptos)**

**Resto de la entrega (corrida, sustentación, etc.) 15%. 10% en la sustentación y 5% se evalúa en el último examen parcial**

## Rúbricas

### Informe de Rendimiento

	Valoración en puntos	Excelente	Competente	Deficiente
Informe: presentación, ortografía y redacción	1	0.75	0.5	< 0.5
Presentación de los datos: datos	2	[2,1]	(1, 0.5]	< 0.5

presentados en tablas y gráficos. Los gráficos están contruidos correctamente.				
Análisis de los resultados: el grupo comenta los resultados de interés y deriva conclusiones cónsonas con estos resultados	2	[2,1]	(1, 0.5]	< 0.5
<b>Total</b>	5			

### Descripción

	Valoración en puntos	Excelente	Competente	Deficiente
Informe: presentación. Ortografía y redacción	1	La presentación del informe es impecable sin problemas de ortografía o redacción	El informe tiene fallas menores en la presentación, ortografía y/o redacción	El informe tiene fallas importantes en la presentación, ortografía y/o redacción
Presentación de los datos: datos presentados en tablas y gráficos.	2	Se grafican todas las variables solicitadas, los gráficos están contruidos correctamente (títulos, unidades, etc.)	Faltan algunos gráficos o, están completos, pero faltan elementos que contribuyen a su comprensión: qué se está graficando, cuáles son las unidades de medida, etc.	No se realizan gráficos o se realizan muy pocos gráficos. La presentación de los gráficos es bastante deficiente.
Análisis de los resultados: el grupo comenta los resultados de interés y deriva conclusiones cónsonas con estos resultados	2	Los estudiantes comentan los resultados obtenidos y presentan conclusiones sobre la arquitectura del sistema y los	Los estudiantes comentan los resultados obtenidos pero no presentan conclusiones sobre la arquitectura del sistema y los	Los estudiantes no comentan los resultados obtenidos o los comentarios son muy deficientes.

		patrones de comunicación.	patrones de comunicación.	
--	--	---------------------------	---------------------------	--

## Funcionamiento del Sistema, Sustentación

Indicador	Valoración en puntos /5	Excelente	Competente	Deficiente
Todos los sensores son implementados correctamente	0.5	0.5	(0.5, 0.25]	< 0.25
Servicios en el PC1: ZeroMQ	0.75	(0.75, 0.5]	(0.5, 0.25]	< 0.25
Servicios PC2: Analítica y Control de semáforos	0.75	(0.75, 0.5]	(0.5, 0.25]	< 0.25
Servicio PC3: Consulta y monitoreo	0.5	0.5	(0.5, 0.25]	< 0.25
Persistencia/actualización de réplicas	0.25	0.25	0.15	< 0.15
Corrida en 3 máquinas	0.75	(0.75, 0.5]	(0.5, 0.25]	< 0.25
Tratamiento de Fallas	0.75	(0.75, 0.5]	(0.5, 0.25]	< 0.25
Código	0.5	0.5	(0.5, 0.25]	< 0.25
Sustentación/Video	0.25	0.25	0.15	< 0.15
Total	5pts			

## Descripción

	Excelente	Competente	Deficiente
Implementación de los procesos/sensores	Los procesos definidos en el enunciado del proyecto son implementados correctamente según las especificaciones del enunciado. Es fácil parametrizar los procesos para cambiar datos como el tamaño de la matriz, tiempo	Hay deficiencias en alguno de estos aspectos: a) Comportamiento que debe tener el proceso. b) Uso del patrón de comunicación recomendado c) parametrización  La mayoría (no todos) de los procesos se implementaron correctamente.	De los 3 aspectos mencionados en la celda anterior, dos o más funcionan de forma incorrecta o incompleta.

	de generación de datos, etc.		
Persistencia/actualización de réplicas	Los campos definidos para la BD son adecuados. Todas las actualizaciones se realizan de forma correcta. La actualización de la réplica se realiza según las indicaciones del enunciado.	Se detectan pequeños problemas en la actualización de datos o de réplicas.	No se implementa la persistencia o no se implementa la réplica.
Corrida en 3 máquinas	El sistema funciona en al menos tres computadoras (o máquinas virtuales) según la arquitectura sugerida en el enunciado del proyecto	El sistema funciona solo en dos computadoras y/o máquinas virtuales	Todos los componentes del sistema se instalan y funcionan en una sola computadora.
Tratamiento de Fallas del PC3	Ante la falla del PC3 todos los procesos se reconectan con la réplica. La falla se detecta de forma automática usando un patrón, como por ejemplo, el health check. Los estudiantes explican el o los patrones de resiliencia implementados	Ante la falla el sistema queda funcionando parcialmente. La falla no se detecta de forma automática. Los estudiantes no explican claramente los patrones de resiliencia implementados.	No se implementa la tolerancia a fallas del servicio en PC3
Código	El código está bien estructurado, sangrado (indentation)	El código es funcional pero presenta alguno de estos problemas: documentación, estructura o sangrado.	El código es deficiente, sin orden ni documentación

	del código correcto y documentado (hay documentación de archivos, funciones o métodos, algoritmos, etc.)		
Sustentación, video	Se siguen todas las reglas dadas para la sustentación, se presentan todos los elementos importantes del sistema y se responden de forma adecuada todas las inquietudes del evaluador. Se entrega el video en el que se evidencia claramente la arquitectura de la aplicación, su funcionamiento y los elementos de tolerancia a fallas.	Hay alguna <i>falla menor</i> en uno o dos de los siguientes elementos: <ul style="list-style-type: none"> <li>- Seguimiento de las reglas dadas para la sustentación</li> <li>- Presentación de las funcionalidades del sistema</li> <li>- Respuesta correcta a todas las preguntas del evaluador.</li> <li>- El video no ilustra de manera suficiente la arquitectura, el funcionamiento o de la aplicación o la tolerancia de las fallas.</li> </ul>	Hay <i>fallas importantes</i> en 3 o más <i>de los</i> siguientes elementos: <ul style="list-style-type: none"> <li>- Seguimiento de las reglas dadas para la sustentación</li> <li>- Presentación de las funcionalidades del sistema</li> <li>- Respuesta correcta a todas las preguntas del evaluador.</li> <li>- El video es deficiente o no se adjunta.</li> </ul>