



**PONTIFICIA UNIVERSIDAD JAVERIANA**  
**FACULTAD DE INGENIERIA**  
**DEPARTAMENTO INGENIERIA DE SISTEMAS**  
Proyecto de Introducción a los Sistemas Distribuidos  
Período Académico 2025-10



## Gestión de aulas

### Objetivos del Proyecto

- 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

## Introducción

En este proyecto, se desarrollará una aplicación que permitirá hacer la administración y asignación de aulas (salones y laboratorios) de una universidad, utilizando conceptos de sistemas distribuidos. Para ello, se requiere el desarrollo de una aplicación distribuida que facilitará la coordinación entre diferentes facultades, garantizando alta disponibilidad y confiabilidad, incluso ante fallos parciales de un servidor que recibirá todas las solicitudes. El sistema a implementar deberá funcionar como mínimo en 3 computadoras (físicas o virtuales).

## Descripción del Problema

En la universidad existen 10 facultades representadas por procesos. Cada facultad recibe solicitudes de aulas por parte de los programas académicos (5 por cada facultad), los cuales también estarán representados por procesos. Los programas académicos se comunicarán con las facultades, quienes a su vez se comunican con un servidor central ubicado en la DTI (representado como un proceso en una de las máquinas).

Se debe implementar un servidor central que recibirá las solicitudes de aulas (laboratorios y salones) de todas las facultades. En el servidor, la solicitud de aulas se maneja mediante la ejecución paralela de múltiples hilos que atienden solicitudes de reserva de forma simultánea, pero asegurando que el acceso a recursos compartidos (en este caso, las aulas) se realice de manera controlada, sin conflictos y asegurando la consistencia de los datos.

Las solicitudes de cada programa académico son variables para cada semestre, pero oscilan entre 7 y 10 aulas (salones + laboratorios), es decir, el número máximo de aulas solicitadas para toda la universidad está entre 350 y 500 aulas. Dentro de este número se solicitan entre 2 y 4 laboratorios por programa, es decir en total se solicitan entre 100 y 200 laboratorios. La universidad cuenta con 380 salones de clase y 60 laboratorios. En caso de que se agoten los laboratorios debido a la alta demanda, existe la posibilidad de adaptar salones como aulas móviles, es decir, se lleva un carrito con portátiles para satisfacer las necesidades de las clases.

Los nombres de las facultades y programas académicos se encuentran al final de este documento.

## Entorno

Tabla 1: Componentes del sistema

DTI	Un servidor central	
Facultades	10	
Programas académicos	5 por cada facultad	50 en total
Solicitudes de aulas (salones de clase + laboratorios)	Número entre 7 y 10 aulas por programa por cada semestre (se recibe como argumento de entrada)	Entre 350 y 500 en total
Solicitudes de Laboratorios	Número entre 2 y 4 por programa por cada semestre	Entre 100 y 200 en total

	(se recibe como argumento de entrada)	
Salones disponibles	380	
Laboratorios disponibles	60	
Total de aulas (Salones + laboratorios)	440	
Máquinas a utilizar	Mínimo 3	

El servidor debe imprimir por pantalla el número de solicitudes que realiza cada facultad (salones y laboratorios), buscar los recursos, realizar la asignación, y enviar a las facultades mensajes con el resultado dicha asignación.

En caso de que no sea posible satisfacer la demanda, el servidor debe generar una alerta indicando que las aulas no son suficientes. Las asignaciones no resultas también se comunican a las Facultades.

## Facultades y programas académicos

En el sistema se crearán procesos para representar las facultades y procesos para representar a los programas académicos. Aunque los programas están asociados a las Facultades, se ejecutarán en computadoras distintas como lo indica la Figura 1. En cada semestre, el sistema debe permitir crear hasta 10 procesos representando a las facultades y 5 procesos representando a los programas académicos asociados a cada Facultad. Los procesos que representan a los programas académicos recibirán como argumentos de entrada: el nombre del programa que se está representando, el semestre, el número salones, el número de laboratorios a solicitar e información que permita comunicarse con su facultad. Los procesos facultad reciben el nombre de la facultad, el semestre y la información necesaria para comunicarse con el servidor. En la figura 1 se representa la ubicación de los componentes principales del sistema en las 3 máquinas:

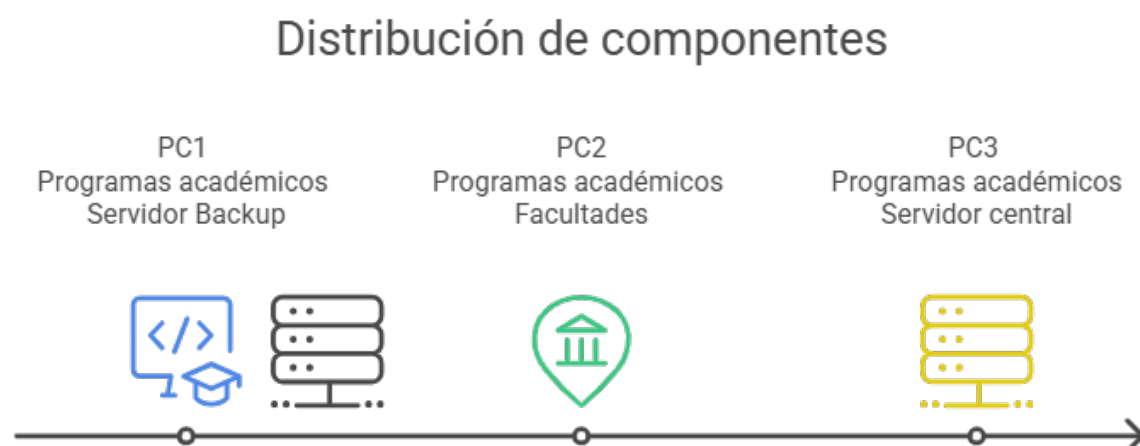


Figura 1: Despliegue de los distintos procesos

Los procesos representando a los programas académicos se comunicarán con las Facultades utilizando un patrón *request-reply*, síncrono de la librería zeroMQ. Una vez que reciban la respuesta, deben almacenarla en un archivo de texto. Cada semestre se genera un archivo distinto.

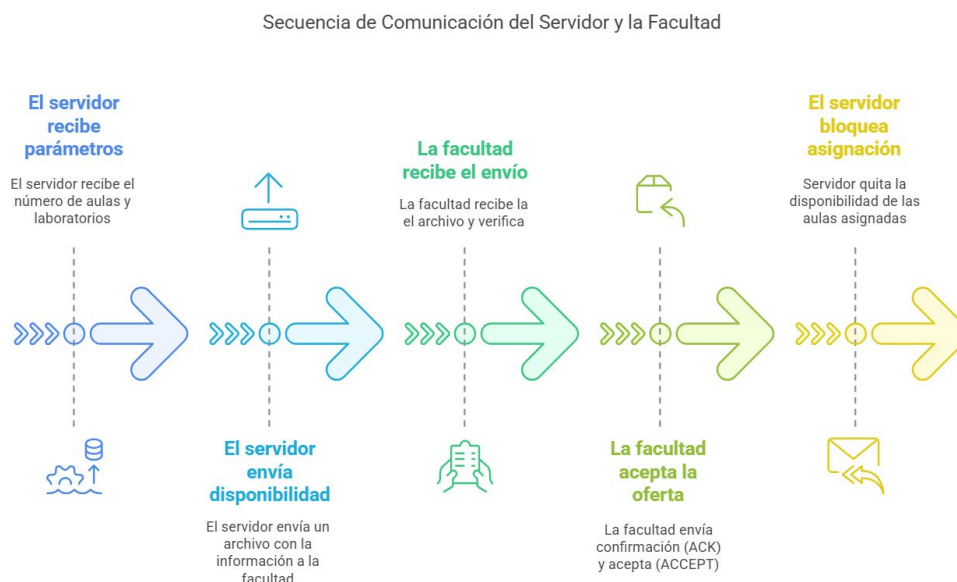


Figura 2. Comunicación entre Facultades y Servidor Central

## Servidor Central

El servidor central es un proceso que tiene las siguientes funciones:

- Recibe las solicitudes las facultades y las atiende de forma concurrente usando hilos. Las solicitudes contienen: semestre, nombre de facultad, nombre y solicitudes de cada programa: número de salones, número de laboratorios
- Asigna los espacios según la disponibilidad.
- Decide si se requiere adaptar un salón como aula móvil, en caso de que se hayan agotado los laboratorios
- Si definitivamente no se puede cumplir con una solicitud, debe generar una alerta y mostrarla por pantalla, pero se debe continuar el proceso. En la respuesta a la Facultad respectiva debe indicar que no se pudo realizar la asignación por falta de recursos. La alerta evidencia que las aulas no son suficientes para responder a la demanda.
- Debe almacenar en un archivo o BD las solicitudes de cada facultad y programa.
- Debe llevar un registro de las solicitudes no atendidas con información de la facultad y programa.
- Para realizar la comunicación entre Facultades y el Servidor va a usar dos patrones de comunicación de la librería zeroMQ: a) **The asynchronous client/server pattern** y b) **load balancing broker** (debe generar 2 versiones de su proyecto, una donde se utilice el patrón a) y

otra donde se utilice el patrón b)). También va a comparar el desempeño de los dos patrones. Más información sobre estos patrones los puede encontrar en

- <https://zguide.zeromq.org/docs/chapter2/>
- <https://zguide.zeromq.org/docs/chapter3/>

El servidor recibe la información que envía cada facultad y responde con las aulas asignadas. **La facultad debe confirmar que recibe y acepta la asignación**

En la figura 3, se representa la interacción y patrón de comunicación entre departamentos, facultades y el servidor central.

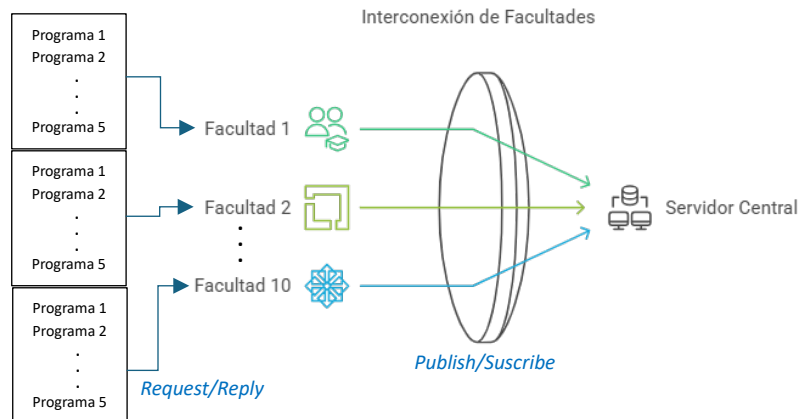


Figura 3. Arquitectura general

## Detalles de Implementación

### Patrones de comunicación:

Como se ha mencionado previamente en este documento, los patrones se implementarán utilizando la librería zeroMQ. Los estudiantes deben respetar los patrones de comunicación indicados en este enunciado. Cualquier dificultad con la implementación de alguno de los patrones, debe comunicarlo al profesor/a o al grupo de apoyo de monitores.

Tabla 2: Patrones de comunicación a implementar

	Patrón
Programas - Facultades	Request-reply síncrono
Facultades-Servidor	Request-reply asíncrono
Facultades-Servidor	Load Balancing Broker

### Patrones de Resiliencia

**Réplica y health-check:** los estudiantes deben diseñar y construir una arquitectura tolerante a fallas, considerando **una posible falla del Servidor Central**. Deberán implementar un respaldo

del proceso servidor que realice las mismas funciones, el cual estará en ubicado una máquina diferente (Dirección IP diferente). Debe existir también otro proceso encargado de chequear que el Servidor está o no en funcionamiento (*health check*). Cuando se detecte que deja de funcionar, el proceso réplica debe asumir la carga. La reconexión del nuevo proceso y distribución de cargas debe hacerse de forma automática, es decir, será transparente para el resto de los procesos del sistema; las operaciones deben continuar en el punto en que se habían dejado y no iniciar desde el principio. Para implementar este patrón (u otro similar, por ejemplo el Heartbeats) pueden revisar <https://zguide.zeromq.org/docs/chapter4/>

## Necesario para la evaluación

El día de la sustentación es importante que se pueda observar:

- Operaciones que va realizando cada uno de los procesos e hilos y el resultado de las operaciones.
- Información guardada en la BD (o en un archivo), tanto en el servir, como en los procesos que representan a los programas académicos.
- El funcionamiento del proyecto distribuyendo los componentes en al menos tres computadoras (o máquinas virtuales): Facultades y programas académicos distribuidas en las tres máquinas, Servidor en una de las tres máquinas y servidor réplica en una máquina diferente (**IP diferente**).

## Rendimiento del Sistema

En este proyecto se va a medir fundamentalmente el tiempo que tarda el servidor en responder a las Facultades, el tiempo que tardan las respuestas en llegar a los programas, y el número de respuestas negativas (porque no hay salones disponibles).

Para la entrega final deberá realizar un **informe de rendimiento del sistema** donde se compararán las implementaciones mencionadas en la sección (Servidor Central). El Informe deberá contener los datos que se piden en la tabla 3. Con estos datos cada grupo realizará gráficos y tablas, y escribirá un análisis sobre los dos patrones utilizados para resolver el problema. Finalmente emitirá conclusiones y recomendaciones.

Tabla 3: Medidas de Rendimiento

	Patrón request-reply asíncrono entre Facultades y Servidor		Patrón Load Balancing Broker entre Facultades y Servidor	
	5 Facultades generando peticiones. 5 Programas	5 Facultades generando peticiones. 5 Programas	5 Facultades generando peticiones. 5 Programas	5 Facultades generando peticiones. 5 Programas

	académicos por Facultad. Cada programa pide los mínimos 7 Aulas y 2 laboratorios (o cómo máximo 2 y 7)	académicos por Facultad. Cada programa pide los máximos 10 Aulas y 4 laboratorios (o cómo máximo 4 y 10)	académicos por Facultad. Cada programa pide los mínimos 7 Aulas y 2 laboratorios (o cómo máximo 2 y 7)	académicos por Facultad. Cada programa pide los máximos 10 Aulas y 4 laboratorios (o cómo máximo 4 y 10)
Tiempo de respuesta promedio (del servidor a las Facultades)				
Tiempo de respuesta mínimo y máximo (del servidor a las Facultades)				
Tiempo promedio desde que los programas hacen los requerimientos hasta que son atendidos				
Por Programa: número de requerimientos que son atendidos de forma satisfactoria				
Por Programas: número de requerimientos que no son atendidos por la Facultad				

### Primera Entrega (15%)

- La primera entrega se realizará en la semana 12 a través de la plataforma. Las sustentaciones serán el día 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 **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.
  - El protocolo de pruebas que utilizará para la entrega final (considere **todos** los tipos de prueba que deben realizarse a un sistema)
  - Cómo va a obtener las métricas de desempeño de la tabla 3: si va a usar herramientas de monitoreo o va a insertar instrucciones especiales en el código. En caso de usar herramientas de monitoreo, cuáles son y cómo las va a usar.
- Implementación inicial:
 

En la primera entrega deben tener implementados los siguientes requerimientos: a) Facultades enviando requerimientos al servidor (solo un patrón de comunicación de los dos exigidos) b) Servidor concurrente acepta los requerimientos, bloquea los salones o laboratorios y responde a las Facultades. c) Las Facultades aceptan los recursos asignados. En esta entrega no es obligatorio implementar los procesos que representan a los programas académicos, los archivos o la tolerancia a fallas. Los estudiantes deben hacer la demostración en al menos dos máquinas (físicas o virtuales).
- Se deben implementar los requerimientos de acuerdo con el enunciado y la rúbrica.
- El día de la sustentación, cada equipo tendrá 10 minutos para mostrar sus resultados y responder las preguntas que tengan los profesores.

## SEGUNDA ENTREGA (15%)

La entrega se realizará en la semana 17 a través de la plataforma. Las sustentaciones se harán los días que correspondan para cada curso. El día de la sustentación los integrantes del equipo deben **mostrar la funcionalidad del proyecto en al menos tres máquinas físicas o virtuales**. Deben estar presentes todos los integrantes del grupo.

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
- Un video de máximo 10 minutos donde muestre:
  - La arquitectura implementada, distribución de los componentes entre máquinas y patrones de comunicación.
  - Tecnologías utilizadas para implementar el sistema y realizar las mediciones.
  - Cómo implementó la arquitectura confiable (tolerante a fallas)
  - Cómo tomó las medidas solicitadas en la tabla 3.
  - Adicionalmente se le puede pedir que traiga el día de la sustentación las diapositivas que usó en su video.



## Informe de Rendimiento (10%)

Como parte de la segunda entrega debe elaborar un informe de máximo 5 páginas donde explican cómo realizaron los experimentos para obtener los datos de la tabla 3, especificaciones de hw y sw donde se realizaron las medidas, variables e instrumentos de medición, resultados obtenidos (acompañados por gráficos y tablas), análisis y conclusiones. El informe se debe entregar en formato pdf.

## Equipos de Trabajo.

- El proyecto debe realizarse en grupos de trabajo de máximo tres personas

## Rúbricas

## Calificación I Entrega

Valor 15%, evaluado sobre 5pts

Indicador	Valoración en puntos /5pts	Excelente	Competente	Deficiente
Informe (presentación, completitud)	0.5	0.5	0.25	<0.25
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	2	[2,1.5]	(1.5, 1.0]	< 1.00
total	5ptos			

Indicador	Excelente	Competente	Deficiente
Informe	La presentación del informe es impecable, sin	El informe tiene fallas menores en la presentación, ortografía	El informe tiene fallas importantes en la presentación,

	problemas de ortografía o redacción. El informe contiene todos los aspectos solicitados	y/o redacción. El informe contiene todos los aspectos solicitados	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	De los siguientes aspectos, solo dos fueron	La implementación inicial es bien

	requeridas en el enunciado para la primera entrega están implementadas correctamente. El sistema funciona en más de un computador (físico o máquina virtual)	presentados correctamente: a) Generación de las posiciones desde los taxis y su recepción por parte del servidor. b) El servidor escoge uno de los taxis y le envía un mensaje indicándole que debe hacer un servicio c) Ejecución en 2 máquinas.	deficiente, se observan falencias en 2 o mas de los de los puntos mencionados en la celda anterior (a-c)
--	--	--	--

## Calificación II Entrega

**Valor 25%**

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

**Resto de la entrega (corrida, sustentación, etc.) 15%. Evaluado sobre 5ptos**

## 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 presentados en tablas y gráficos. Los gráficos están contruidos correctamente.	2	[2,1]	(1, 0.5]	< 0.5
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

	<b>Valoración en puntos</b>	<b>Excelente</b>	<b>Competente</b>	<b>Deficiente</b>
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 patrones de comunicación.	Los estudiantes comentan los resultados obtenidos pero no presentan conclusiones sobre la arquitectura del sistema y los patrones de comunicación.	Los estudiantes no comentan los resultados obtenidos o los comentarios son muy deficientes.

### Funcionamiento del Sistema, Sustentación

Indicador	<b>Valoración en puntos /30</b>	<b>Excelente</b>	<b>Competente</b>	<b>Deficiente</b>
Programas académicos	0.5	0.5	0.25	< 0.25
Facultades	0.5	0.5	0.25	< 0.25

Servidor concurrente	0.75	[0.75, 0.5]	(0.5, 0.25]	< 0.25
Patrones de comunicación	0.5	0.5	0.25	< 0.25
Persistencia	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 del Servidor-	0.5	0.5	0.25	< 0.25
Código	0.25	0.25	(0.25, 0.15]	< 0.15
Sustentación, video	1	[1,0.75]	(0.75, 0.5]	< 0.5
Total	5pts			

## Descripción

	<b>Excelente</b>	<b>Competente</b> -	<b>Deficiente</b>
Programas académicos	Los procesos que representan a los programas académicos son implementados correctamente según las especificaciones del enunciado.	Hay deficiencias en alguno de estos aspectos (máximo uno): a) Argumentos de entrada que reciben los procesos que representan a los taxis periódicamente b) Comportamiento que debe tener el proceso. b) Uso del patrón d comunicación recomendado	De los 3 aspectos mencionados en la celda anterior, dos o mas funcionan de forma incorrecta o incompleta.
Facultades	Los procesos que representan a las facultades son implementados correctamente según las especificaciones del enunciado.	Hay deficiencias en alguno de estos aspectos (máximo uno): a) Argumentos de entrada que reciben los procesos que representan a los taxis periódicamente b) Comportamiento que debe tener el proceso. b) Uso del patrón d comunicación recomendado	De los 3 aspectos mencionados en la celda anterior, dos o mas funcionan de forma incorrecta o incompleta.

Servidor concurrente	El servidor es implementado correctamente. Se utilizan hilos para favorecer la concurrencia. Se generan las alarmas cuando es necesario. La asignación de salones, laboratorios y aulas móviles se asigna de forma correcta.	El servidor funciona, pero no es concurrente (no tiene hilos para realizar las diversas tareas). O bien el servidor es concurrente, pero se observan fallas menores en su funcionamiento. Se generan las alarmas. Pueden observarse pequeños problemas en la asignación de salones y laboratorios.	El servidor no funciona correctamente.
Patrones de comunicación	Los estudiantes implementan de forma correcta todos los patrones de comunicación sugeridos. Los estudiantes son capaces de explicar el funcionamiento de los patrones.	El proyecto funciona, pero no se usaron todos los patrones sugeridos. Los estudiantes tienen problemas en explicar alguno de los patrones de comunicación	La comunicación dentro del proyecto funciona incorrectamente.
Persistencia	Se almacena toda la información indicada en el enunciado.	Se almacena aproximadamente el 80% de la información indicada en el enunciado	No se almacena información sobre las entidades representadas en el proyecto, o la información almacenada es escasa.
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 Servidor	Ante la falla del proxy la inicialización de un proceso homólogo funciona de forma correcta. La falla se detecta de forma automática usando un patrón, como por ejemplo, el health check. Los estudiantes	Ante la falla del proxy se levanta un proceso homólogo que asume de forma correcta sus funciones. La falla no se detecta de forma automática. Los estudiantes no explican claramente los patrones	No se implementa la tolerancia a fallas de este componente.

	explican el o los patrones de resiliencia implementados	de resiliencia implementados.	
Código	El código está bien estructurado, sangrado (indentation) del código correcto y documentado (hay documentación de archivos, funciones o métodos, algoritmos, etc)	El código es funcional pero presenta alguno de estos problemas: documentación, estructura o sangrado.	El código es deficiente, sin orden ni documentación
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 de la aplicación o la tolerancia de las fallas.</li> </ul>	Hay <i>fallas</i> importantes 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>

## Estructura de Facultades y Programas

### Facultad de Ciencias Sociales

- Programa de Psicología
- Programa de Sociología

- Programa de Trabajo Social
- Programa de Antropología
- Programa de Comunicación

## Facultad de Ciencias Naturales

- Programa de Biología
- Programa de Química
- Programa de Física
- Programa de Geología
- Programa de Ciencias Ambientales

## Facultad de Ingeniería

- Programa de Ingeniería Civil
- Programa de Ingeniería Electrónica
- Programa de Ingeniería de Sistemas
- Programa de Ingeniería Mecánica
- Programa de Ingeniería Industrial

## Facultad de Medicina

- Programa de Medicina General
- Programa de Enfermería
- Programa de Odontología
- Programa de Farmacia
- Programa de Terapia Física

## Facultad de Derecho

- Programa de Derecho Penal
- Programa de Derecho Civil
- Programa de Derecho Internacional
- Programa de Derecho Laboral
- Programa de Derecho Constitucional

## Facultad de Artes

- Programa de Bellas Artes
- Programa de Música
- Programa de Teatro
- Programa de Danza
- Programa de Diseño Gráfico



## Facultad de Educación

- Programa de Educación Primaria
- Programa de Educación Secundaria
- Programa de Educación Especial
- Programa de Psicopedagogía
- Programa de Administración Educativa

## Facultad de Ciencias Económicas

- Programa de Administración de Empresas
- Programa de Contabilidad
- Programa de Economía
- Programa de Mercadotecnia
- Programa de Finanzas

## Facultad de Arquitectura

- Programa de Arquitectura
- Programa de Urbanismo
- Programa de Diseño de Interiores
- Programa de Paisajismo
- Programa de Restauración de Patrimonio

## Facultad de Tecnología

- Programa de Desarrollo de Software
- Programa de Redes y Telecomunicaciones
- Programa de Ciberseguridad
- Programa de Inteligencia Artificial
- Programa de Big Data