

Evaluación Parcial N° 2

Desarrollo y configuración de microservicios caso semestral.

Estudainte

Sigla	Nombre Asignatura	Tiempo Asignado	% Ponderación
DSY1104	DESARROLLO FULLSTACK I	1 SEMANA	35%

1. Situación evaluativa 1

	Ejecución práctica
--	-----------------------

X	Entrega de encargo
---	-----------------------

	Presentación
--	--------------

2.Situación evaluativa 2

	Ejecución práctica
--	-----------------------

	Entrega de encargo
--	-----------------------

X	Presentación
---	--------------

3. Instrucciones generales para el/la estudiante

- **Descripción general de la evaluación:** Cada evaluación parcial será una parte del examen transversal, dividiendo el caso en tres partes para las tres evaluaciones parciales. En cada evaluación parcial se evaluará el avance del caso, centrado en el logro de los indicadores de logro relacionados con el RA. En esta evaluación, los/las estudiantes avanzarán en el desarrollo y configuración de los microservicios propuestos en la evaluación parcial 1. Esta etapa del caso semestral involucra los RA2 y RA3, centrándose en la implementación práctica de los conceptos relacionados con la configuración de proyectos con componentes de microservicios y el desarrollo de componentes de backend.
- **Tiempo** asignado para esta evaluación es de **7 horas pedagógicas** y se realiza en **equipos** (máximo 3 estudiantes por equipo) en **taller de alto cómputo**. Sin embargo, **la evaluación de la presentación será individual**.
- La **distribución de los porcentajes de las situaciones evaluativas que componen esta evaluación** es la siguiente:

Evaluación	Porcentaje dentro de la asignatura	Tipo de situación evaluativa	Distribución de porcentajes
Evaluación Parcial N° 2	35%	Entrega de encargo	40%
		Presentación	60%

Instrucciones específicas de la Evaluación:

Situación evaluativa 1: Entrega por encargo

En el informe, los estudiantes deben:

1. Diseñar un proyecto utilizando el framework Spring, demostrando comprensión de su estructura y funcionalidades básicas.
 - a. Gestionar las dependencias y el ciclo de vida del proyecto de microservicios usando Maven, mostrando habilidades en la configuración de archivos POM y en la ejecución de comandos de Maven.
 - b. Construir componentes backend del proyecto, definiendo una línea base de trabajo clara, demostrando habilidades en diseño y desarrollo de microservicios.
2. Configurar un proyecto de microservicios utilizando herramientas y frameworks específicos (como Spring y Maven).
 - a. Emplear el control de versiones con Git, incluyendo la configuración inicial del repositorio y el seguimiento de cambios en el código.

- b. Demostrar una comprensión de la arquitectura interna de los microservicios y cómo estos se integran en un entorno de desarrollo fullstack, explicando la estructura y la interacción entre los diferentes componentes.
3. El estudiante debe demostrar la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, asegurando que el código sea modular, mantenible y escalable.
 - a. El estudiante debe mostrar competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, configurando y gestionando correctamente los proyectos.
 - b. El estudiante debe demostrar habilidades de trabajo colaborativo utilizando Git, manejando correctamente el control de versiones, integraciones y revisiones de código para asegurar la funcionalidad y calidad del código generado.
4. Implementar operaciones CRUD utilizando tecnologías del framework de backend (por ejemplo, JPA, ORM) y garantiza su funcionamiento con datos provenientes de Postman.
 - a. Implementar comunicación RESTful entre diferentes componentes de microservicio.
5. Validar la comunicación entre microservicios usando Postman, probando y consumiendo los servicios del backend para asegurar que funcionen correctamente.

El informe debe incluir los siguientes apartados:

Instrucciones:

A. Desarrollo de Microservicios:

- *Implementar cada uno de los microservicios identificados en la fase de diseño.*
- *Asegurar la comunicación entre los microservicios mediante APIs RESTful.*

B. Base de Datos:

- *Configurar y gestionar bases de datos que soporten cada uno de los microservicios.*
- *Asegurar la consistencia y la integridad de los datos.*

C. Integración de Servicios:

- *Desarrollar mecanismos para la integración continua y entrega continua (CI/CD) de los microservicios.*
- *Configurar un entorno de prueba que permita la validación de los microservicios de manera aislada y en conjunto.*

D. Pruebas Unitarias:

- *Implementar pruebas unitarias para cada microservicio.*
- *Documentar los resultados de las pruebas.*

E. Entregables:

- *Código fuente de los microservicios desarrollados.*
- *Configuración de bases de datos.*
- *Documentación de la integración y entrega continua.*
- *Resultados de las pruebas unitarias.*

Situación evaluativa 2: Presentación

Durante la presentación, cada estudiante debe:

1. Presentar el diseño del proyecto utilizando Spring. Debe demostrar comprensión de la estructura del framework y sus funcionalidades básicas, explicando cómo se ha aplicado en el proyecto.
2. Explicar cómo utilizó Maven para gestionar las dependencias y el ciclo de vida del proyecto. Debe mostrar conocimiento en la configuración de archivos POM y la ejecución de comandos de Maven, proporcionando ejemplos específicos.
3. Detallar la línea base de trabajo establecida y cómo se construyeron los componentes backend. Debe demostrar habilidades en diseño y desarrollo de microservicios, explicando los pasos y decisiones clave en el proceso.
4. Describir cómo configuró el proyecto utilizando herramientas y frameworks específicos (como Spring y Maven). Debe explicar el uso del control de versiones con Git, incluyendo la configuración inicial del repositorio y el seguimiento de cambios en el código.
5. Demostrar comprensión de la arquitectura interna de los microservicios y cómo estos se integran en un entorno de desarrollo fullstack. Debe explicar la estructura de los microservicios y la interacción entre los diferentes componentes del sistema.

6. Exponer sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes backend. Debe explicar cómo aseguraron que el código fuera modular, mantenible y escalable, proporcionando ejemplos específicos.
7. Presentar el diseño del proyecto utilizando Spring. Debe demostrar comprensión de la estructura del framework y sus funcionalidades básicas, explicando cómo se ha aplicado en el proyecto.
8. Mostrar competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend. Debe explicar la configuración y gestión de proyectos, detallando cómo utilizaron estas herramientas en su proyecto.
9. Demostrar habilidades de trabajo colaborativo utilizando Git. Debe explicar cómo manejaron el control de versiones, integraciones y revisiones de código, asegurando la funcionalidad y calidad del código generado.
10. Describir el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend (como JPA y ORM) para realizar operaciones CRUD. Debe explicar cómo validaron estas operaciones con datos provenientes de Postman.
11. Explicar la implementación de comunicación RESTful entre diferentes componentes de microservicio. Debe asegurar que la integración sea eficiente y coherente, proporcionando detalles técnicos y ejemplos.
12. Mostrar cómo validó la comunicación entre microservicios utilizando Postman. Debe explicar cómo probaron y consumieron los servicios del backend para asegurar que funcionen correctamente.

2. Pauta de Evaluación

Tipo de Pauta: Rúbrica

Categoría	% logro	Descripción niveles de logro
Muy buen desempeño	100%	Demuestra un desempeño destacado, evidenciando el logro de todos los aspectos evaluados en el indicador.
Buen desempeño	80%	Demuestra un alto desempeño del indicador, presentando pequeñas omisiones, dificultades y/o errores.
Desempeño aceptable	60%	Demuestra un desempeño competente, evidenciando el logro de los elementos básicos del indicador, pero con omisiones, dificultades o errores.
Desempeño incipiente	30%	Presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador, por lo que no puede ser considerado competente.
Desempeño no logrado	0%	Presenta ausencia o incorrecto desempeño.

Indicador de Evaluación	Categorías de Respuesta					Ponderación Indicador de Evaluación
	Muy buen desempeño 100%	Buen desempeño 80%	Desempeño aceptable 60%	Desempeño incipiente 30%	Desempeño no logrado 0%	
SITUACIÓN EVALUATIVA 1: ENCARGO. INFORME CON EVALUACIÓN GRUPAL.						
IE2.1.1. Diseña un proyecto utilizando herramientas y frameworks específicos para microservicios (como Spring, Maven, entre otros), demostrando habilidades en la definición de una línea base de trabajo y la construcción de piezas de backend.	Diseña correctamente un proyecto utilizando herramientas y frameworks específicos para microservicios (como Spring, Maven, entre otros), demostrando habilidades en la definición de una línea base de trabajo y la construcción de piezas de backend, respondiendo a las necesidades del caso propuesto.	Diseña un proyecto utilizando herramientas y frameworks específicos para microservicios, demostrando habilidades en la definición de una línea base de trabajo y la construcción de piezas de backend, pero presentando pequeñas omisiones, dificultades y/o errores.	Diseña un proyecto utilizando herramientas y frameworks específicos para microservicios, demostrando habilidades en la definición de una línea base de trabajo y la construcción de piezas de backend, pero presenta omisiones, dificultades o errores.	Diseña un proyecto utilizando herramientas y frameworks específicos para microservicios, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No diseña un proyecto utilizando herramientas y frameworks específicos para microservicio, o lo hace de forma incorrecta.	6%

<p>IE2.2.1. Configura un proyecto de microservicios utilizando herramientas y frameworks seleccionados, ocupando control de versiones Git, incluyendo la comprensión de la arquitectura interna de un microservicio y su integración en un entorno de desarrollo fullstack.</p>	<p>Configura correctamente un proyecto de microservicios utilizando herramientas y frameworks seleccionados, ocupando control de versiones Git, incluyendo la comprensión de la arquitectura interna de un microservicio y su integración en un entorno de desarrollo fullstack.</p>	<p>Configura un proyecto de microservicios utilizando herramientas y frameworks seleccionados, ocupando control de versiones Git, incluyendo la comprensión de la arquitectura interna de un microservicio y su integración en un entorno de desarrollo fullstack, pero presentando pequeñas omisiones, dificultades y/o errores.</p>	<p>Configura un proyecto de microservicios utilizando herramientas y frameworks seleccionados, ocupando control de versiones Git, pero presenta omisiones, dificultades o errores.</p>	<p>Configura un proyecto de microservicios utilizando herramientas y frameworks, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.</p>	<p>No configura un proyecto de microservicios, o lo hace de forma incorrecta.</p>	<p>6%</p>
<p>IE2.3.1: Desarrolla componentes de backend dentro de un proyecto de microservicios, aplicando buenas prácticas de diseño y arquitectura, utilizando herramientas como Spring y Maven, trabajando colaborativamente con Git, garantizando la funcionalidad del código generado.</p>	<p>Desarrolla componentes de backend dentro de un proyecto de microservicios, aplicando buenas prácticas de diseño y arquitectura, utilizando herramientas como Spring y Maven, trabajando colaborativamente con Git, garantizando la funcionalidad del código generado de forma correcta y coherente.</p>	<p>Desarrolla componentes de backend dentro de un proyecto de microservicios, aplicando buenas prácticas de diseño y arquitectura, utilizando herramientas como Spring y Maven, trabajando colaborativamente con Git, pero presentando pequeñas omisiones, dificultades y/o errores.</p>	<p>Desarrolla componentes de backend dentro de un proyecto de microservicios, aplicando buenas prácticas de diseño y arquitectura, pero presenta omisiones, dificultades o errores.</p>	<p>Desarrolla componentes de backend dentro de un proyecto de microservicios, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.</p>	<p>No desarrolla componentes de backend dentro de un proyecto de microservicios, o lo hace de forma incorrecta.</p>	<p>14%</p>

IE3.1.1: Desarrolla componentes de microservicio utilizando tecnologías del framework de backend (como JPA, ORM, etc.) para realizar operaciones CRUD con datos provenientes de Postman.	Desarrolla correctamente componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman.	Desarrolla componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, pero presentando pequeñas omisiones, dificultades y/o errores.	Desarrolla componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, pero presenta omisiones, dificultades o errores.	Desarrolla componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No desarrolla componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, o lo hace de forma incorrecta.	6%
IE3.2.1: Implementa comunicación RESTful entre componentes de microservicio, asegurando una integración coherente entre ellos, y utilizando herramientas como Postman para validar y consumir servicios del backend.	Implementa correctamente comunicación RESTful entre componentes de microservicio, asegurando una integración coherente entre ellos, y utilizando herramientas como Postman para validar y consumir servicios del backend.	Implementa comunicación RESTful entre componentes de microservicio, utilizando herramientas como Postman para validar y consumir servicios del backend, pero presentando pequeñas omisiones, dificultades y/o errores.	Implementa comunicación RESTful entre componentes de microservicio, utilizando herramientas como Postman para validar y consumir servicios del backend, pero presenta omisiones, dificultades o errores.	Implementa comunicación RESTful entre componentes de microservicio, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No implementa comunicación RESTful entre componentes de microservicio, o lo hace de forma incorrecta.	8%
Total Situación evaluativa 1						40%

SITUACIÓN EVALUATIVA 2: PRESENTACIÓN. DESEMPEÑO INDIVIDUAL.

IE2.1.2: Responde las 2 preguntas formuladas por el docente, dando cuenta que domina el diseño del proyecto de microservicios utilizando Spring, demostrando comprensión de su estructura y funcionalidades básicas.	Responde las 2 preguntas formuladas por el docente, dando cuenta que domina el diseño del proyecto de microservicios utilizando Spring, demostrando comprensión de su estructura y funcionalidades básicas.	Responde las 2 preguntas formuladas por el docente, pero presentando pequeñas omisiones, dificultades y/o errores.	Responde las preguntas formuladas por el docente, pero presenta omisiones, dificultades o errores.	Responde las preguntas formuladas por el docente, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No responde las preguntas formuladas por el docente, o lo hace de forma incorrecta.	3%
IE2.1.3: Explica el uso de Maven para gestionar las dependencias y el ciclo de vida del proyecto, mostrando conocimiento en la configuración de archivos POM y la ejecución de comandos de Maven.	Explica el uso de Maven para gestionar las dependencias y el ciclo de vida del proyecto de forma detallada y correcta, mostrando conocimiento en la configuración de archivos POM y la ejecución de comandos de Maven.	Explica el uso de Maven mostrando conocimiento en la configuración de archivos POM y la ejecución de comandos de Maven, pero presentando pequeñas omisiones, dificultades y/o errores.	Explica el uso de Maven mostrando conocimiento en la configuración de archivos POM y la ejecución de comandos de Maven, pero presenta omisiones, dificultades o errores.	Explica el uso de Maven mostrando conocimiento en la configuración de archivos POM y la ejecución de comandos de Maven, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No explica el uso de Maven, o lo hace de forma incorrecta.	3%

IE2.1.4: Detalla la línea base de trabajo y la construcción de componentes backend, demostrando habilidades en diseño y desarrollo de microservicios.	Detalla correctamente la línea base de trabajo y la construcción de componentes backend, demostrando habilidades en diseño y desarrollo de microservicios.	Detalla la línea base de trabajo y la construcción de componentes backend, pero presentando pequeñas omisiones, dificultades y/o errores.	Detalla la línea base de trabajo y la construcción de componentes backend, pero presenta omisiones, dificultades o errores.	Detalla la línea base de trabajo y la construcción de componentes backend, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No detalla la línea base de trabajo y la construcción de componentes backend, o lo hace de forma incorrecta.	3%
IE2.2.2: Describe la configuración del proyecto de microservicios utilizando herramientas y frameworks específicos (como Spring y Maven) y el uso del control de versiones con Git.	Describe la configuración del proyecto de microservicios utilizando herramientas y frameworks específicos y el uso del control de versiones con Git de forma detallada y correcta.	Describe la configuración del proyecto de microservicios utilizando herramientas y frameworks específicos y el uso del control de versiones con Git, pero presentando pequeñas omisiones, dificultades y/o errores.	Describe la configuración del proyecto de microservicios utilizando herramientas y frameworks específicos y el uso del control de versiones con Git, pero presenta omisiones, dificultades o errores.	Describe la configuración del proyecto de microservicios utilizando herramientas y frameworks específicos y el uso del control de versiones con Git, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No describe la configuración del proyecto de microservicios, o lo hace de forma incorrecta.	3%

IE2.2.3: Demuestra comprensión de la arquitectura interna de los microservicios y su integración en un entorno de desarrollo fullstack, explicando la estructura y la interacción entre los diferentes componentes.	Demuestra comprensión de la arquitectura interna de los microservicios y su integración en un entorno de desarrollo fullstack, explicando correctamente la estructura y la interacción entre los diferentes componentes.	Demuestra comprensión de la arquitectura interna de los microservicios y su integración en un entorno de desarrollo fullstack, explicando la estructura y la interacción entre los diferentes componentes, pero presentando pequeñas omisiones, dificultades y/o errores.	Demuestra comprensión de la arquitectura interna de los microservicios y su integración en un entorno de desarrollo fullstack, explicando la estructura y la interacción entre los diferentes componentes, pero presenta omisiones, dificultades o errores.	Demuestra comprensión de la arquitectura interna de los microservicios y su integración en un entorno de desarrollo fullstack, explicando la estructura y la interacción entre los diferentes componentes, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No demuestra comprensión de la arquitectura interna de los microservicios y su integración en un entorno de desarrollo fullstack, o lo hace de forma incorrecta.	3%
IE2.3.2 Expone sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, asegurando que el código sea modular, mantenible y escalable.	Expone de forma correcta y detallada sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, asegurando que el código sea modular, mantenible y escalable.	Expone sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, pero presentando pequeñas omisiones, dificultades y/o errores.	Expone sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, pero presenta omisiones, dificultades o errores.	Expone sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No expone sobre la aplicación de buenas prácticas de diseño y arquitectura en el desarrollo de componentes de backend, o lo hace de forma incorrecta.	3%
IE2.3.3: Responde las 2 preguntas formuladas	Responde en forma correcta y detallada las	Responde las 2 preguntas formuladas	Responde las preguntas formuladas	Responde las preguntas formuladas	No Responde las preguntas	9%

por el docente, dando cuenta que domina el diseño del proyecto de microservicios utilizando Spring, demostrando comprensión de su estructura y funcionalidades básicas	2 preguntas formuladas por el docente, dando cuenta que domina el diseño del proyecto de microservicios utilizando Spring, demostrando comprensión de su estructura y funcionalidades básicas	por el docente, pero presentando pequeñas omisiones, dificultades y/o errores.	por el docente, pero presenta omisiones, dificultades o errores.	por el docente, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	formuladas por el docente, o lo hace de forma incorrecta.	
IE2.3.4: Muestra competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, explicando la configuración y gestión de proyectos.	Muestra competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, explicando la configuración y gestión de proyectos.	Muestra competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, pero presentando pequeñas omisiones, dificultades y/o errores.	Muestra competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, pero presenta omisiones, dificultades o errores.	Muestra competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No muestra competencia en el uso de herramientas y frameworks como Spring y Maven para construir componentes de backend, o lo hace de forma incorrecta.	9%

IE2.3.5: Demuestra habilidades de trabajo colaborativo utilizando Git, explicando el manejo correcto del control de versiones, integraciones y revisiones de código para asegurar la funcionalidad y calidad del código generado.	Demuestra habilidades de trabajo colaborativo utilizando Git, explicando el manejo correcto del control de versiones, integraciones y revisiones de código para asegurar la funcionalidad y calidad del código generado.	Demuestra habilidades de trabajo colaborativo utilizando Git, pero presentando pequeñas omisiones, dificultades y/o errores.	Demuestra habilidades de trabajo colaborativo utilizando Git, pero presenta omisiones, dificultades o errores.	Demuestra habilidades de trabajo colaborativo utilizando Git, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No demuestra habilidades de trabajo colaborativo utilizando Git, o lo hace de forma incorrecta.	3%
IE3.1.2: Describe el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend (por ejemplo, JPA, ORM) para realizar operaciones CRUD con datos provenientes de Postman.	Describe de forma correcta y detallada el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman.	Describe el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, pero presentando pequeñas omisiones, dificultades y/o errores.	Describe el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, pero presenta omisiones, dificultades o errores.	Describe el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No Describe el desarrollo de componentes de microservicio utilizando tecnologías del framework de backend para realizar operaciones CRUD con datos provenientes de Postman, o lo hace de forma incorrecta.	9%

IE3.2.2: Explica la implementación de comunicación RESTful entre diferentes componentes de microservicio, asegurando una integración eficiente y coherente.	Explica de forma correcta y detallada la implementación de comunicación RESTful entre diferentes componentes de microservicio, asegurando una integración eficiente y coherente.	Explica la implementación de comunicación RESTful entre diferentes componentes de microservicio, pero presentando pequeñas omisiones, dificultades y/o errores.	Explica la implementación de comunicación RESTful entre diferentes componentes de microservicio, pero presenta omisiones, dificultades o errores.	Explica la implementación de comunicación RESTful entre diferentes componentes de microservicio, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No explica la implementación de comunicación RESTful entre diferentes componentes de microservicio, o lo hace de forma incorrecta.	9%
IE3.2.3: Muestra cómo validó la comunicación entre microservicios usando Postman, probando y consumiendo los servicios del backend.	Muestra de forma correcta y detallada cómo validó la comunicación entre microservicios usando Postman, probando y consumiendo los servicios del backend.	Muestra cómo validó la comunicación entre microservicios usando Postman, probando y consumiendo los servicios del backend, pero presentando pequeñas omisiones, dificultades y/o errores.	Muestra cómo validó la comunicación entre microservicios usando Postman, probando y consumiendo los servicios del backend, pero presenta omisiones, dificultades o errores.	Muestra cómo validó la comunicación entre microservicios usando Postman, probando y consumiendo los servicios del backend, pero presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador.	No muestra cómo validó la comunicación entre microservicios usando Postman, o lo hace de forma incorrecta.	3%
Total Situación evaluativa 2						60%
Total						100%