Evaluación Parcial N° 3

Integrando backend y frontend a la tienda online

Docente

Sigla	Nombre Asignatura	Tiempo Asignado	% Ponderación
DSY1104	DESARROLLO FULLSTACK II	1 SEMANA	40%

1. Situación evaluativa 1

	Ejecución práctica	X	Entrega de encargo		Presentación

2.Situación evaluativa 2

	Ejecución práctica		Entrega de encargo	X	Presentación

3. Tabla de Especificaciones

Resultado de Aprendizaje	Indicador de Logro (IL)	Ponderación Indicador de Logro	Indicador de Evaluación (IE)	Ponderación Indicador de Evaluación	N° de pregunta o ítems
SITUACIÓN EVALUATIVA 1: ENCARGO. INFORME CON EVALUACIÓN GRUPAL.					
▪ **RA3: Integra componente front con componente backend, usando comunicación REST, para obtener información desde bases de datos.**	▪ IL3.1. Crea una aplicación web utilizando un framework backend con conexión a base de datos, considerando los requerimientos establecidos por el cliente.	8%	IE3.1.1 Crea una aplicación backend con conexión a una base de datos, implementando lógica de negocio y modelos de datos, acorde a los requerimientos establecidos por el cliente.	8%	1
	▪ IL3.2. Implementa la integración entre servicios backend y frontend, mediante la comunicación REST, realizando operaciones de lectura, creación, actualización y eliminación de registros.	14%	IE3.2.1 Implementa en el backend servicios de API REST utilizando Spring Boot, incorporando endpoints para realizar operaciones de lectura, creación, actualización y eliminación de registros, mostrando los endpoint en Swagger.	8%	2
			IE3.2.2 Implementa integración entre servicios de backend y frontend mediante la comunicación de la API REST.	6%	3

▪ **RA3: Integra componente front con componente backend, usando comunicación REST, para obtener información desde bases de datos.**	▪ IL3.3. Establece el acceso a la aplicación web mediante la autenticación y autentificación segura de los datos del cliente y restringiendo el acceso a funcionalidades propias del usuario.	18%	IE3.3.1: Implementa autenticación de usuarios en el backend utilizando roles para asegurar que solo los usuarios autorizados puedan acceder a ciertos recursos, utilizando la autentificación basada en tokens (JWT).	6%	**4**
			IE3.3.2: Desarrolla un sistema de gestión de sesiones en el frontend que mantenga el estado de autenticación del usuario de manera segura, permitiendo la persistencia de la sesión incluso en caso de recargas de página.	6%	5
			IE3.3.3: Desarrolla restricciones de acceso a funcionalidades en el frontend, asegurando que las interfaces y acciones sean accesibles únicamente por usuarios con los permisos adecuados.	6%	6
Total Situación evaluativa 1		**40%**		**40%**	

SITUACIÓN EVALUATIVA 2: PRESENTACIÓN. DESEMPEÑO INDIVIDUAL.					
▪ **RA3: Integra componente front con componente backend, usando comunicación REST, para obtener información desde bases de datos.**	▪ IL3.1. Crea una aplicación web utilizando un framework backend con conexión a base de datos, considerando los requerimientos establecidos por el cliente.	12%	IE3.1.2 Describe el desarrollo de una aplicación backend que se conecta a una base de datos, implementando lógica de negocio y modelos de datos según los requerimientos del cliente.	12%	1
	▪ IL3.2. Implementa la integración entre servicios backend y frontend, mediante la comunicación REST, realizando operaciones de lectura, creación, actualización y eliminación de registros.	22%	IE3.2.3 Explica cómo se implementaron servicios de API REST utilizando Spring Boot, incorporando endpoints para realizar operaciones CRUD y documentando en Swagger.	12%	2
			IE3.2.4 Justifica la integración efectiva entre servicios de backend y frontend mediante la comunicación de la API REST, asegurando un flujo de datos eficiente.	10%	3

	▪ IL3.3. Establece el acceso a la aplicación web mediante la autenticación y autentificación segura de los datos del cliente y restringiendo el acceso a funcionalidades propias del usuario.	26%	IE3.3.4: Describe la implementación de autenticación de usuarios utilizando roles y autenticación basada en tokens (JWT) para asegurar el acceso seguro a recursos.	10%	4
			IE3.3.5: Expone el desarrollo de un sistema de gestión de sesiones en el frontend para mantener el estado de autenticación de los usuarios de manera segura.	8%	5
			IE3.3.6: Explica cómo se desarrollaron restricciones de acceso a funcionalidades en el frontend, asegurando que solo los usuarios con permisos adecuados puedan acceder a ciertas interfaces y acciones.	8%	6
Total Situación evaluativa 2		**60%**		**60%**	
Total EFT		**100%**		**100%**	

4.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 RA3. Tras realizar la evaluación parcial 2, los estudiantes seguirán trabajando en su desarrollo semestral. El equipo deberá integrar su frontend con un backend en Spring, documentar y desarrollar pruebas de integración, asegurar la correcta comunicación y seguridad entre frontend y backend.
- Tiempo asignado para esta evaluación es de 1 semana 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° 3	40%	Entrega de encargo	40%
		Presentación	60%

- Orientaciones para la implementación de la evaluación
  - *Las instrucciones se entregarán en la semana 16 y la entrega en la semana 17.*
  - *El trabajo debe ser desarrollado en Taller de alto cómputo.*
  - *El informe debe ser cargado en AVA en la fecha correspondiente.*
  - *Tras realizar la evaluación parcial 2, los estudiantes seguirán trabajando en su desarrollo semestral.*
  - *La situación evaluativa 1, denominada Entrega de encargo, representa la culminación del trabajo colaborativo realizado por el equipo. En esta fase, es fundamental que la evaluación sea equitativa entre todos los integrantes del grupo.*
  - *La situación evaluativa 2, la Presentación, constituye una oportunidad para que los estudiantes defiendan el proyecto desarrollado por sus equipos. Durante esta etapa, es esencial evaluar individualmente a cada estudiante, ya sea mediante su participación en la exposición del proyecto o a través de las respuestas dadas a las preguntas formuladas por el docente. Es crucial que tanto la justificación como la argumentación presentada sean consistentes con el proyecto y pertinentes a la asignatura en cuestión.*

Instrucciones específicas de la Evaluación:

Situación evaluativa 1: Entrega por encargo

En la propuesta de desarrollo web, los estudiantes deben:

1. Crea una aplicación backend con conexión a una base de datos, implementando lógica de negocio y modelos de datos, acorde a los requerimientos establecidos por el cliente.
    a. Confirma que el backend se conecte correctamente a la base de datos y que el modelo de datos refleje los requisitos del cliente.
    b. Asegúrate de que la lógica de negocio implementada resuelva los problemas planteados por los requisitos del cliente.
    c. Describe el desarrollo de la aplicación backend que se conecta a una base de datos, explicando cómo implementaste la lógica de negocio y los modelos de datos según los requerimientos del cliente.
2. Implementa en el backend servicios de API REST utilizando Spring Boot, incorporando endpoints para realizar operaciones de lectura, creación, actualización y elminacion de registros, mostrando los endpoint en Swagger.
    a. Revisa que los endpoints del API REST estén implementados correctamente y realicen las operaciones CRUD requeridas, mostrando los endpoints esté disponible en Swagger.
    b. Explica cómo implementaste servicios de API REST en el backend utilizando Spring Boot. Detalla los endpoints creados para realizar operaciones CRUD y cómo documentaste estos endpoints en Swagger.
3. Implementa integracion entre servicios de backend y frontend mediante la comunicación de la API REST.
    a. Comprueba que los servicios de backend y frontend se comuniquen correctamente.
    b. Justifica cómo lograste una integración efectiva entre los servicios de backend y frontend mediante la API REST. Explica el flujo de datos entre ambos componentes y cómo aseguraste una comunicación eficiente.
4. Implementa autenticación de usuarios en el backend utilizando roles para asegurar que solo los usuarios autorizados puedan acceder a ciertos recursos, utilizando la autentificación basada en tokens (JWT).
    a. Asegúrate de que el sistema de autenticación gestione adecuadamente los roles y permisos de los usuarios.
    b. Verifica que la generación, verificación y renovación de tokens JWT se realice de manera segura.
    c. Describe cómo implementaste la autenticación de usuarios en el backend utilizando roles y autenticación basada en tokens (JWT). Explica el proceso de asegurar el acceso a recursos y cómo gestionaste la autenticación de manera segura.
5. Desarrolla un sistema de gestión de sesiones en el frontend que mantenga el estado de autenticación del usuario de manera segura, permitiendo la persistencia de la sesión incluso en caso de recargas de página.
    a. Confirma que la sesión del usuario se mantenga activa y segura, incluso en caso de recargas de página.
    b. Expone el desarrollo del sistema de gestión de sesiones en el frontend para mantener el estado de autenticación de los usuarios. Detalla cómo aseguraste la persistencia y seguridad del estado de sesión, incluso en caso de recargas de página.
6. Desarrolla restricciones de acceso a funcionalidades en el frontend, asegurando que las interfaces y acciones sean accesibles únicamente por usuarios con los permisos adecuados.

a. Comprueba que las restricciones de acceso en el frontend funcionen correctamente según los roles de usuario.
b. Explica cómo desarrollaste restricciones de acceso a funcionalidades en el frontend, asegurando que solo los usuarios con permisos adecuados puedan acceder a ciertas interfaces y acciones. Detalla el enfoque utilizado para gestionar permisos y cómo esto contribuye a la seguridad y funcionalidad del proyecto. Proporciona ejemplos de interfaces restringidas.

Documentos requeridos del proyecto por el grupo:

Entregables:

- *Enlace GitHub público del proyecto frontend.*
- *Enlace GitHub público del proyecto backend.*
- *Proyecto frontend comprimido.*
- *Proyecto backend comprimido.*
- *Manual de usuario (con pantallazos detallados).*
- *Documento de integración.*

Situación evaluativa 2: Presentación

Durante la presentación, cada estudiante debe:

1. Describe el desarrollo de una aplicación backend que se conecta a una base de datos, implementando lógica de negocio y modelos de datos según los requerimientos del cliente.

   Preguntas sugeridas: Realizar 3 preguntas específicas a cada integrante. ¿Cómo configuraste la conexión a la base de datos y qué tipo de base de datos usaste?, ¿Qué modelos de datos creaste para tu aplicación y cómo se relacionan con los requerimientos del cliente?, ¿Puedes describir cómo implementaste la lógica de negocio en tu aplicación backend?

2. Explica cómo se implementaron servicios de API REST utilizando Spring Boot, incorporando endpoints para realizar operaciones CRUD y documentando en Swagger.

   Preguntas sugeridas: Realizar 3 preguntas específicas a cada integrante. ¿Puedes describir cómo implementaste la lógica de negocio en tu aplicación backend?, ¿Qué estrategia utilizaste para documentar tus endpoints con Swagger y por qué es importante esta documentación?, ¿Cómo manejaste la comunicación entre el backend y el frontend para asegurar un flujo de datos eficiente?

3. Justifica la integración efectiva entre servicios de backend y frontend mediante la comunicación de la API REST, asegurando un flujo de datos eficiente.

   Preguntas sugeridas: Realizar 3 preguntas específicas a cada integrante. ¿Qué mecanismos de control de errores implementaste en tus endpoints para manejar respuestas fallidas y cómo estas se comunican al frontend?, ¿Puedes explicar cómo implementaste la paginación o la gestión de grandes volúmenes de datos en uno de tus endpoints para optimizar el rendimiento de la API?, ¿Puedes explicar cómo es el flujo de un crud, desde su vista en el frontend y consulta en el backend?

4. IE3.3.4: Describe la implementación de autenticación de usuarios utilizando roles y autenticación basada en tokens (JWT) para asegurar el acceso seguro a recursos.

   Preguntas sugeridas: Realizar 3 preguntas específicas a cada integrante. ¿Cómo verificas y valida el token JWT en el backend antes de conceder acceso a los recursos protegidos? Muestra un fragmento de código donde esto se realiza, ¿Cómo defines y asignas roles en tu aplicación, y cómo afecta esto la generación de tokens JWT? Proporciona un ejemplo de código, ¿Puedes mostrar cómo manejas la expiración de tokens JWT y cómo respondes cuando un token ha caducado?

5. IE3.3.5: Expone el desarrollo de un sistema de gestión de sesiones en el frontend para mantener el estado de autenticación de los usuarios de manera segura.

   Preguntas sugeridas: Realizar 3 preguntas específicas a cada integrante. ¿Cómo implementas el almacenamiento de tokens JWT en el frontend, y qué medidas tomas para proteger este almacenamiento?, ¿Puedes explicar cómo detectas y gestionas el estado de sesión en caso de recarga de página o cierre de navegador? Proporciona un ejemplo de código, ¿Cómo manejas la actualización automática o manual de tokens en el frontend para mantener una sesión activa sin interrupciones?

6. IE3.3.6: Explica cómo se desarrollaron restricciones de acceso a funcionalidades en el frontend, asegurando que solo los usuarios con permisos adecuados puedan acceder a ciertas interfaces y acciones.

   Preguntas sugeridas: Realizar 3 preguntas específicas a cada integrante. ¿Cómo implementas la lógica de comprobación de roles en el frontend para mostrar u ocultar interfaces según los permisos del usuario?, ¿Puedes mostrar un ejemplo de cómo utilizas rutas protegidas en tu aplicación para controlar el acceso a ciertas partes del frontend? ¿Cómo aseguras que las acciones sensibles del usuario, como enviar datos críticos, están adecuadamente protegidas y verificadas en el frontend?

5. 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.						
IE3.1.1 Crea una aplicación backend con conexión a una base de datos, implementando lógica de negocio y modelos de datos, acorde a los requerimientos establecidos por el cliente.	Crea una aplicación backend perfectamente conectada a una base de datos, implementando lógica de negocio y modelos de datos de forma excepcional.	Crea una aplicación backend competente con pequeñas omisiones en la lógica de negocio o modelos de datos.	Crea una aplicación backend con varios errores o deficiencias en la lógica de negocio o modelos de datos.	Crea una aplicación backend deficiente con errores graves en la lógica de negocio o modelos de datos.	No crea una aplicación backend funcional o correctamente conectada a una base de datos.	8%
IE3.2.1 Implementa en el backend servicios de API REST utilizando Spring Boot, incorporando endpoints para realizar operaciones de lectura, creación, actualización y eliminación de registros, mostrando los endpoint en Swagger.	Implementa servicios de API REST en el backend utilizando Spring Boot, con endpoints funcionales para todas las operaciones CRUD, documentados en Swagger.	NO APLICA	Implementa servicios de API REST, pero con algunos endpoints que no funcionan correctamente o que están mal documentados.	NO APLICA	No implementa servicios de API REST efectivos o correctamente documentados en el backend.	8%

IE3.2.2 Implementa integración entre servicios de backend y frontend mediante la comunicación de la API REST.	Justifica de manera convincente la integración efectiva entre servicios de backend y frontend mediante la comunicación de la API REST, asegurando un flujo de datos eficiente.	Justifica bien la integración con algunos detalles menores omitidos sobre la eficiencia del flujo de datos.	Justifica la integración adecuadamente pero con omisiones claras en la eficiencia del flujo de datos.	Justificación insuficiente de la integración con errores notables en la eficiencia del flujo de datos.	No justifica adecuadamente la integración o el flujo de datos entre el backend y el frontend.	6%
IE3.3.1: Implementa autenticación de usuarios en el backend utilizando roles para asegurar que solo los usuarios autorizados puedan acceder a ciertos recursos, utilizando la autentificación basada en tokens (JWT).	Implementa autenticación de usuarios en el backend utilizando roles y autenticación basada en tokens (JWT) de manera que asegure un acceso seguro a los recursos.	NO APLICA	Implementa autenticación de usuarios, pero con algunos errores en la gestión de roles o en la seguridad de los tokens.	NO APLICA	No implementa autenticación de usuarios de manera efectiva o segura en el backend.	6%
IE3.3.2: Desarrolla un sistema de gestión de sesiones en el frontend que mantenga el estado de autenticación del usuario de manera segura, permitiendo la persistencia de la sesión incluso en caso de recargas de página.	Desarrolla un sistema de gestión de sesiones en el frontend que mantiene el estado de autenticación de manera perfecta, permitiendo la persistencia de la sesión.	Desarrolla un sistema de gestión de sesiones competente con pequeñas omisiones en la persistencia o seguridad de la sesión.	Desarrolla un sistema de gestión de sesiones con varios errores o deficiencias en la persistencia o seguridad.	Desarrolla un sistema de gestión de sesiones de manera deficiente con errores graves en la persistencia o seguridad.	No desarrolla un sistema de gestión de sesiones efectivo o seguro en el frontend.	6%

IE3.3.3: Desarrolla restricciones de acceso a funcionalidades en el frontend, asegurando que las interfaces y acciones sean accesibles únicamente por usuarios con los permisos adecuados.	Desarrolla restricciones de acceso a funcionalidades en el frontend de manera perfecta, asegurando que solo usuarios autorizados accedan a interfaces específicas.	Desarrolla restricciones de acceso competentes con pequeñas omisiones en la seguridad o funcionalidad de las restricciones.	Desarrolla restricciones de acceso con varios errores o deficiencias en la seguridad o funcionalidad.	Desarrolla restricciones de acceso de manera deficiente con errores graves en la seguridad o funcionalidad.	No desarrolla restricciones de acceso efectivas o adecuadas en el frontend.	6%
					Total Situación evaluativa 1	40%
SITUACIÓN EVALUATIVA 2: PRESENTACIÓN. DESEMPEÑO INDIVIDUAL.						
IE3.1.2 Describe el desarrollo de una aplicación backend que se conecta a una base de datos, implementando lógica de negocio y modelos de datos según los requerimientos del cliente.	Describe de manera ejemplar el desarrollo de una aplicación backend conectada a una base de datos, implementando lógica de negocio y modelos de datos perfectos.	Describe bien el desarrollo de una aplicación backend con algunas pequeñas omisiones en la lógica de negocio o modelos de datos.	Describe el desarrollo de una aplicación backend pero con errores notables en la lógica de negocio o modelos de datos.	Describe de manera deficiente el desarrollo con errores graves en la lógica de negocio o modelos de datos.	No describe adecuadamente el desarrollo de una aplicación backend conectada a una base de datos.	12%
IE3.2.3 Explica cómo se implementaron servicios de API REST utilizando Spring Boot, incorporando endpoints para realizar operaciones CRUD y documentando en Swagger.	Explica de manera ejemplar cómo se implementaron servicios de API REST utilizando Spring Boot, con endpoints perfectamente funcionales para operaciones CRUD.	Explica bien cómo se implementaron servicios con algunos detalles menores omitidos en la funcionalidad o documentación de los endpoints.	Explica cómo se implementaron servicios con varias omisiones o errores en la funcionalidad o documentación de los endpoints.	Explicación insuficiente con errores graves en la funcionalidad o documentación de los servicios de API REST.	No explica adecuadamente cómo se implementaron servicios de API REST o la funcionalidad de los endpoints.	12%

IE3.2.4 Justifica la integración efectiva entre servicios de backend y frontend mediante la comunicación de la API REST, asegurando un flujo de datos eficiente.	Justifica correctamente la integración efectiva entre servicios de backend y frontend mediante la comunicación de la API REST, con un análisis exhaustivo del flujo de datos.	Justifica bien la integración con algunos detalles menores omitidos en el análisis del flujo de datos.	Justifica la integración adecuadamente pero con omisiones claras en el análisis del flujo de datos.	Justificación insuficiente de la integración con errores notables en el análisis del flujo de datos.	No justifica adecuadamente la integración o el análisis del flujo de datos entre el backend y el frontend.	10%
IE3.3.4: Describe la implementación de autenticación de usuarios utilizando roles y autenticación basada en tokens (JWT) para asegurar el acceso seguro a recursos.	Describe de manera ejemplar la implementación de autenticación de usuarios utilizando roles y autenticación basada en tokens (JWT), asegurando un acceso seguro.	Describe bien la implementación de autenticación con algunos detalles menores omitidos en la gestión de roles o tokens.	Describe la implementación de autenticación pero con errores notables en la gestión de roles o tokens.	Describe de manera deficiente la implementación con errores graves en la gestión de roles o tokens.	No describe adecuadamente la implementación de autenticación de usuarios en el backend.	10%
IE3.3.5: Expone el desarrollo de un sistema de gestión de sesiones en el frontend para mantener el estado de autenticación de los usuarios de manera segura.	Expone correctamente el desarrollo de un sistema de gestión de sesiones en el frontend, asegurando la persistencia y seguridad de la sesión.	NO APLICA	Expone bien el desarrollo de un sistema de gestión de sesiones con algunos detalles menores omitidos en la persistencia o seguridad.	NO APLICA	No expone de manera adecuada el desarrollo de un sistema de gestión de sesiones en el frontend.	8%

IE3.3.6: Explica cómo se desarrollaron restricciones de acceso a funcionalidades en el frontend, asegurando que solo los usuarios con permisos adecuados puedan acceder a ciertas interfaces y acciones.	Explica de manera ejemplar cómo se desarrollaron restricciones de acceso a funcionalidades en el frontend, asegurando que solo usuarios autorizados accedan.	Explica bien el desarrollo de restricciones de acceso con algunos detalles menores omitidos en la seguridad o funcionalidad de las restricciones.	Explica adecuadamente el desarrollo de restricciones de acceso pero con omisiones claras en la seguridad o funcionalidad.	Explica de manera insuficiente el desarrollo con errores notables en la seguridad o funcionalidad de las restricciones.	No explica adecuadamente el desarrollo de restricciones de acceso en el frontend.	8%
					Total Situación evaluativa 2	60%
					Total	100%