

PROYECTO FINAL DE CURSO

Sección	:	CURS-000231
Curso	:	Microservices Architecture: EDA, CQRS, SAGA & Kafka (Java, Spring Boot & Spring Cloud)
Instructor	:	Aristedes Novoa Arbildo
Horario	:	
Alumno	:	

I. CONSIDERACIONES GENERALES

- El desarrollo del proyecto es personal, sin embargo, está permitido colaborar con sus compañeros de clase o la asesoría de cualquier otro profesional con experiencia en desarrollo de arquitecturas de microservicios utilizando Java y Spring Cloud u otros stacks tecnológicos.
- Está permitido usar libros, tutoriales, scripts y presentaciones para revisar y repasar conceptos y codificación a reutilizar.
- Reutilizar proyectos previos desarrollados en clase o descargados de internet, incluyendo librerías y rutinas de código; siempre que no violen la propiedad de derecho del autor, en caso de reutilizar códigos o arquitecturas mantener los créditos del autor(es).

II. ACTIVIDADES A REALIZAR

- Elaborar el modelo general (diagrama de organización y dependencia de los microservicios y tecnologías involucradas) de los principales microservicios a desarrollar, considerando microservicios de negocio, gestión de la arquitectura (servidor de configuración, servidor de descubrimiento, API Gateway, etc.), seguridad, monitoreo entre otros; además de los microservicios exclusivos de implementación del patrón CQRS y SAGA. Se recomienda acompañar prototipos de alto nivel a nivel de mockups, diagramas de contexto o flujo de procesos para contextualizar la solución a implementar.
- Crear un repositorio en github, gitlab o cualquier otra plataforma de gestión de versiones para versionar el código fuente de su solución arquitectónica.
- La solución debe incluir los principales componentes, considerando entre los más importantes los siguientes:

#	Componente	Descripción	Cumple ¹		Comentarios de revisión
			Si	No	
1	Diseño conceptual y técnico (2 puntos)	Diseño conceptual de alto nivel y detallado de la arquitectura de microservicios implementando los patrones CQRS y SAGA. Instalar y configurar la infraestructura a utilizar, considerando servidores de bases de datos relacionales (RDBMS - SQL), No SQL , Cache (Redis), Kafka (Zookeeper o Karft); entre otros.			
2	Arquitectura base (2 puntos)	Crear una arquitectura base considerando los siguientes patterns: decompose by businss capability, externalized configuration (server config), service registry (server registry), circuit breaker (reliability), monitoring (server admin) y API Gateway. Puede utilizar docker, docker-compose o kubernetes para remplazar algunos microservicios como es el caso de server config y server registry.			
3	CQRS (5 puntos)	Implementar el patrón CQRS incluyendo el uso de bases de datos relacionales (RDBMS - SQL), un message brocker (Kafka), Redis y bases de datos No SQL (MongoDB), registro, consultas con y sin filtros y actualización de documentos específicos. Implementar resiliencia (re-intentos, re-encolados) y consistencia.			
4	SAGA (5 puntos)	Implementar el patrón SAGA- Orquestado (recomendado) incluyendo el uso de bases de datos relacionales (RDBMS - SQL), un message brocker (Kafka), Redis . Considerar un proceso de negocio que incluye cálculos, validaciones, transformaciones, aprobaciones de manera automática, semi-automática y manual. Implementar resiliencia (re-intentos, re-encolados) y consistencia.			

5	Event (Domain – Sourcing) (3 puntos)	Implementar el uso de Domain Events y Event Sourcing con Spring Events u otra tecnológica, de preferencia utilizar programación reactiva (RxJava, Web Flux) .			
6	Otros (3 puntos)	Implementar otros patrones de microservicios y cualquier otra mejora que complemente su proceso de investigación y entrenamiento en diseño e implementación de arquitecturas de microservicios.			
	Total 20 puntos				

III. PRESENTACION

El alumno deberá realizar la presentación de su proyecto con la implementación de las funcionalidades solicitadas en la fecha establecida por la Coordinación Académica de Galaxy Training.