

Capa	Componente	Descripción Componente	Patrones de Diseño	Tecnología	Descripción tecnología
Interfaz (web)	Dtos	Gestiona los datos ingresados por el cliente para realizar las peticiones HTTP	Data Transfer Object	Java	
Interfaz (web)	Controllers	Gestiona las solicitudes HTTP para dirigir las peticiones al caso de uso correspondiente	Controlador	SpringBoot	
Infraestructura (infrastructure)	Adapters	Adaptan las entidades para que puedan ser utilizadas por los casos de uso. Permiten la integración con componentes externos	Adaptadores	SpringBoot	
Infraestructura	Entities	Objetos de base de datos	Data Access Object	SpringBoot	
Infraestructura (infrastructure)	Repositories	Interfaces para la persistencia de datos	Repository Pattern	Spring Data JPA	
Aplicación (application)	Interfaces	Interfaces para habilitar el consumo de los repositorios para los casos de uso	Strategy	SpringBoot	
Aplicación (application)	Services	Logica de negocio específica para una funcionalidad	Servicios	SpringBoot	
Dominio (domain)	Entities	Entidades de negocio	Domain Driven Desing	SpringBoot	
Dominio (domain)	UseCases	Logica de negocio específica para una funcionalidad	Servicios Interactor	SpringBoot	
	Pruebas Unitarias	Permite verificar la funcionalidad individual de la	Unit Testing	JUnit	
	Servicio de autenticación	Maneja la autenticación y autorización mediante el uso de OAuth2.0.	Service layer	Spring Security	
	Configuración de seguridad	Configura en el cliente los datos requeridos para la autenticación y autorización	Configuration Pattern	SpringBoot	

	Vista Controller	Maneja la lógica de comunicación en el front end	API Gateway Seguridad Configuración	Angular	Flexibilidad Arquitectura modular Permite el desarrollo de pruebas unitarias Escalable Mantenible
	Manejador de colas	Maneja la comunicación asincrona	Productor/Consumidor (Message Queue)	RabbitMQ Kafka	RabbitMQ es un agente que permite gestionar datos de diferentes orígenes y distribuirlo a diferentes destinos. Kafka es una plataforma que cuenta con un sistema de gestión de mensajes escalable y
	Contenedor de aplicaciones	Permita gestionar la ejecución de la aplicación en un ambiente aislado	Contenedor (Containerization)	Docker	
	Base de Datos	Almacen de datos de la aplicación	Base de datos relacional	Postgres AWS RDS S3	Trabajar con tecnologías en la nube que permitan manejar de manera adecuada la escalabilidad y el almacenamiento de información, esto considerando mejores niveles de seguridad y administración de acceso
	Integración Continua	Automatiza el proceso de puesta en producción y prueba del software	Integración Continua (CI/CD)	Azure DevOps Jenkins Sonar	Uso de Azure pipelines para el despliegue y gestión de recursos para el escalamiento de la aplicación. (Pago) Jenkins permite el despliegue de las aplicaciones y gestionamiento de versiones de manera gratuita

	Despliegue en Nube	Brinda el acceso a infraestructura que permite desplegar y escalar la aplicación	Despliegue en la nube	AWS/Azure	Depende de las ventajas de cada plataforma respecto a costos y conocimiento dentro del equipo que la va a administrar
	Monitorización	Permite supervizar el comportamiento de la aplicación y registrar eventos importantes o críticos	Observability	Grafana Kibana CloudWatch Azure Monitor	