Informe EduTech Innovators SPA

**Antonio Valdes | Desarrollo FullStack**

INTRODUCCIÓN AL PROBLEMA

EduTech Innovators SPA es una empresa chilena emergente enfocada en la creación y distribución de plataformas educativas en línea. Debido a su crecimiento exponencial y la expansión de sus operaciones, la empresa enfrenta problemas de rendimiento y disponibilidad derivados de su actual sistema monolítico. El presente informe busca proponer una solución basada en arquitectura de microservicios para superar estas limitaciones

ANÁLISIS DE REQUERIMIENTOS

**Requerimientos Funcionales:**

* **Gestor de Usuarios**: Creación, modificación y eliminación de cuentas
* **Configuración de Permisos de Acceso**: Asignación de roles y privilegios
* **Monitoreo del Sistema y gestión de alertas**: Registro de actividad, generación de alertas
* **Respaldo y Restauración de Datos**: Backup automático y manual
* **Gestor de Cursos**: Creación, administración y publicación de contenido educativo
* **Administración de Instructores**: Asignación de cursos, evaluación de contenidos
* **Gestor de Incidencias**: Manejo de soporte técnico y consultas
* **Inscripción y Seguimiento de Cursos**: Para estudiantes y profesionales
* **Facturación y Pagos**: Procesamiento de pagos y suscripciones

**Requerimientos No Funcionales:**

* **Escalabilidad**: Adaptación a un crecimiento constante de usuarios
* **Alta Disponibilidad**: Garantizar acceso ininterrumpido
* **Seguridad**: Protección de datos y cumplimiento de normativas
* **Mantenibilidad**: Modularidad para facilitar mejoras y actualizaciones

**Análisis de Necesidades de los Usuarios:**

Para conocer mejor las necesidades de los distintos tipos de usuarios, se realizaron entrevistas y encuestas a administradores, vendedores y clientes de la plataforma

**Necesidades según el tipo de usuario**:

**Administradores**: Necesitan controlar los permisos de los usuarios, ver la actividad del sistema en tiempo real y contar con herramientas simples para hacer respaldos y recuperar datos

**Vendedores**: Buscan una forma rápida de registrar solicitudes, seguir el estado de ventas y pagos, y generar reportes de facturación de manera sencilla

**Clientes**: Quieren que el proceso de inscripción sea fácil, que la navegación por la plataforma sea clara, y que el soporte técnico responda de forma rápida y efectiva

ANÁLISIS DEL SISTEMA ACTUAL

El sistema actual de EduTech Innovators SPA es de tipo monolítico, lo que significa que todas sus funciones están unidas en una sola aplicación. Esto ha generado varios problemas debido al rápido crecimiento de la empresa

**Los principales problemas que se presentan son**:

**Falta de escalabilidad**: El sistema no está preparado para manejar más usuarios y nuevas funcionalidades de forma eficiente

**Baja disponibilidad**: Hay caídas frecuentes del sistema, lo que afecta a los usuarios

**Lentitud en horarios de alta demanda**: Cuando muchos usuarios se conectan al mismo tiempo, el sistema se vuelve lento

**Dificultad para actualizar**: Cualquier cambio o mejora en el sistema requiere mucho tiempo y puede afectar otras partes del sistema

**Problemas con el crecimiento**: El sistema no se adapta bien a la expansión de la empresa, como la apertura de nuevas oficinas y el aumento de usuarios

Estos problemas han generado una mala experiencia para los usuarios y complicaciones para el equipo técnico. Por eso, se considera necesario cambiar a una arquitectura más moderna y flexible.

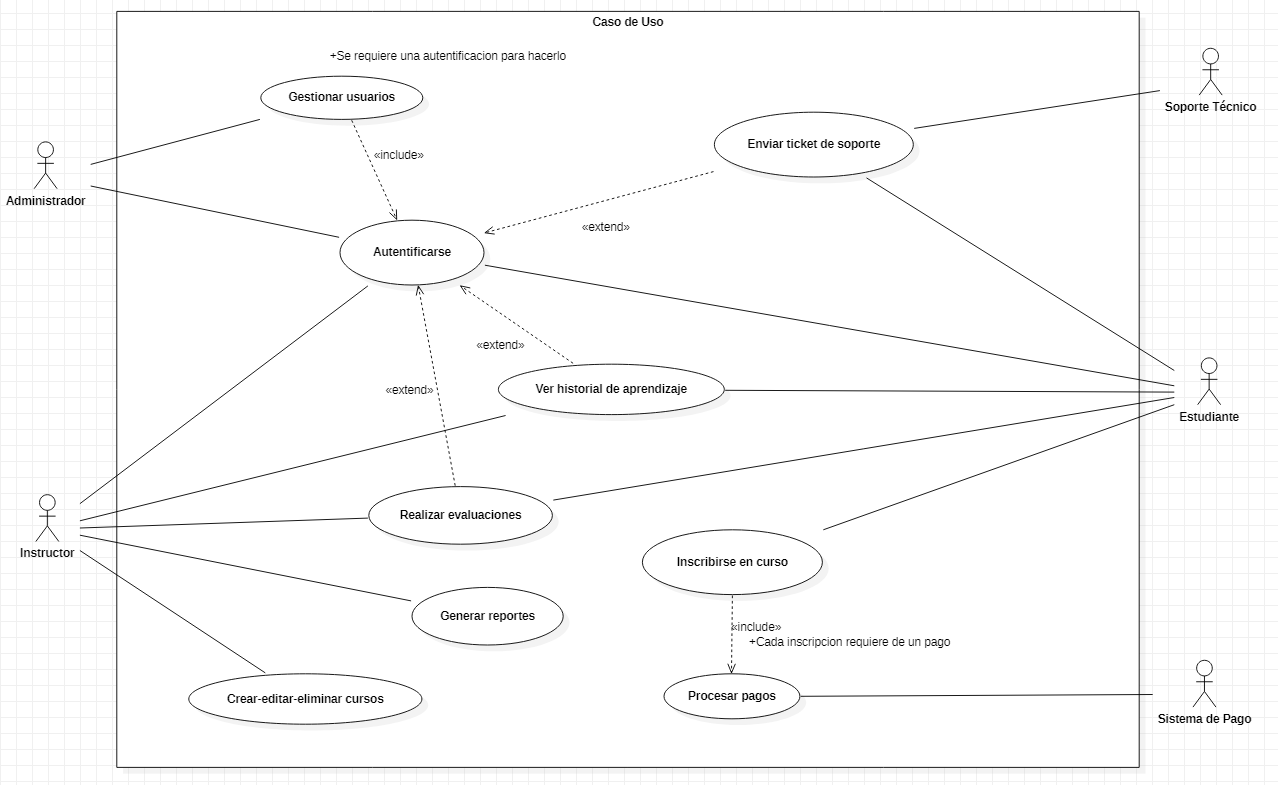
DISEÑO DE LA NUEVA ARQUITECTURA

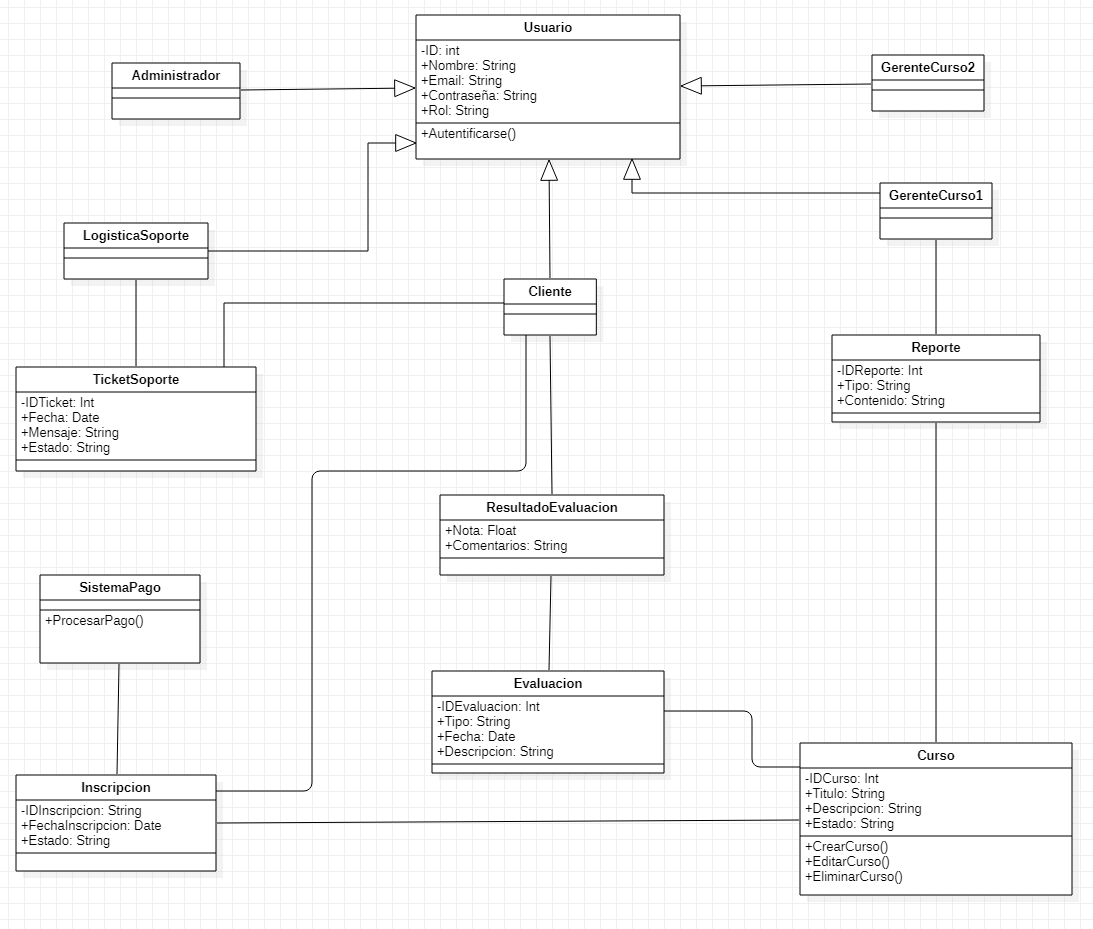
Se propone reemplazar el sistema actual por una arquitectura basada en microservicios. Esto permitirá que cada parte del sistema funcione de forma independiente, facilitando el mantenimiento, el crecimiento y la estabilidad

**Servicios principales que se deben desarrollar:**

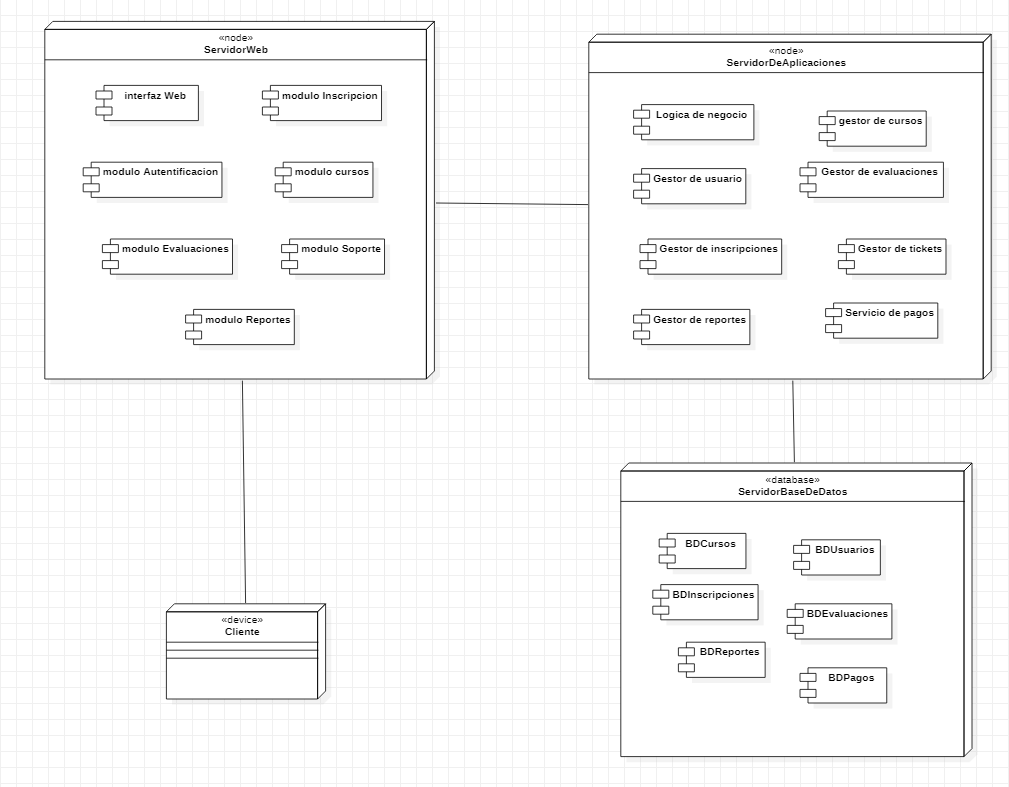
* **Autenticación y autorización**: Se encargará del inicio de sesión, validación de usuarios y control de accesos según el tipo de perfil
* **Gestión de cursos**: Permitirá administrar el contenido educativo, como agregar, modificar o eliminar cursos
* **Evaluaciones y calificaciones**: Permitirá crear pruebas, recibir respuestas de los usuarios y mostrar resultados o retroalimentación
* **Soporte técnico**: Permitirá a los usuarios reportar problemas y hacer seguimiento de sus solicitudes
* **Facturación y pagos**: Se encargará de procesar pagos de manera segura y generar comprobantes o facturas

**Diagramas de la Arquitectura**

* Diagrama de Caso de Uso:
* Diagrama de Clases:



* Diagrama de Despliegue:



ESTRATEGIA DE MITIGACIÓN

**La transición se realizará por fases:**

* Implementación del servicio de autenticación
* Migración del gestor de cursos a microservicios
* Despliegue incremental de los servicios restantes

**Planificación de la Migración:**

* **Evaluación del sistema actual**: Identificar y analizar los módulos del sistema monolítico
* **Definición de microservicios**: Dividir el sistema en servicios independientes (ej. usuarios, cursos, pagos, soporte)
* **Desarrollo de APIs**: Cada servicio se comunicará a través de su propia API
* **Migración progresiva**: Implementar y probar cada servicio por separado antes de eliminar el componente monolítico
* **Pruebas continuas**: Validar el funcionamiento en cada etapa
* **Capacitación del equipo**: Asegurar que todos comprendan y puedan operar con la nueva arquitectura

PLAN DE MITIGACIÓN DE RIESGOS

| **RIESGO** | **DESCRIPCIÓN** | **PLAN DE MITIGACIÓN** |
| --- | --- | --- |
| Interrupción del servicio | Fallos durante el proceso de migración | Migración por etapas y pruebas antes del despliegue |
| Compatibilidad con sistemas heredados | Incompatibilidades con funciones antiguas | Uso de APIs de integración y entornos de prueba |
| Seguridad en la transición de datos | Pérdida o exposición de información sensible | Encriptación AES-256 y autenticación multifactor |
| Rollback y contingencia | Necesidad de volver a la versión anterior en caso de fallos | Mantenimiento de copias y entornos paralelos |
| Falta de experiencia en microservicios | El equipo puede no estar familiarizado con esta arquitectura | Capacitaciones técnicas y mentoría externa |
| Problemas de comunicación entre servicios | Mal funcionamiento por errores en integración | Uso de gateways API y herramientas como RabbitMQ o Kafka |
| Costos de infraestructura más altos | Más servidores y servicios implican mayores costos | Escalamiento progresivo en la nube y ajuste de recursos por demanda |
| Consistencia de datos | Posibles diferencias entre servicios al manejar datos relacionados | Diseño adecuado de sincronización y transacciones distribuidas |