Plan action

Vamos a trabajar con Daniel, ahora daniel presenta la primera actividad del plan action.

Desde hoy hasta el primero de noviembre estaran enfocadas en fortalecer algunas debilidades, performance, comentario que han tenido con los trainers para poder mejorar y fortalecer el conocimiento que tenemos acerca de lo que es diseñar sistemas, desarrollar, han visto problemas con lo que es el diseño de sistemas, lo primero que hay que hacer es diseñar, si no se tiene un buen diseño no tendrás un norte. Entonces han planificado una práctica que se va a llevar adelante para movernos en las sesiones 1 a 1, en las sesiones grupales que vayamos a tener, de manera que todos estemos alineados en un solo tema, esta práctica está relacionada a la arquitectura la idea es que como equipo y de manera personal diseñemos un sistema para administrar un colegio. Cuando ingresamos a un proyecto el produyct owner va a tener un grupo de historias agrupadas en el product backlog, estas historias estan basadas en un documento de requerimientos y alcances que el po ha generado en reunións constantes con el stake holder, en el sprint planning se nos dara una vista de lo que se espera que entreguemos, en el documento esta detallado lo que quieren que desarrollemos, un sistema para administrar un colegio, student enrollment by class, subject and class offerings, grades per student. Basado en la informacion del documento tenemos que diseñar y modelar el dominio, que es muy distinto a lo que es el modelo, los dtos, etc. Debemos construir diagramas apropiados y documentacion que le haga saber al equipo en que estamos trabajando, cual es el alcance del sistema? La administracion de materias, se debe poder administrar clases, las cuales contienen materias, que se dan en una gestion, el primer grado de la gestion 2025 tiene como profesor al licenciado A , otro modulo es la inscripcion de estudiantes, se debe poder inscribir y eliminar estudiantes, considerando la capacidad. Se debe tener u modulo que se pérmita asignar notas a los estudiantes por clase, se debe poder visualizar o listar a los estudiantes y a sus notas, consideren LA AUDITORIA AL MOMENTO DE INGRESAR Y ACTUALIZAR las notas, los estudiantes deben poder ingresar al sistema y ver su progreso.

El sistema debe ser escalabale, debe ser seguro, RBAC, auditability, integridad de datos (se debe prevenir el enrollmente cuando un curso tiene20 estudiantes) las notas deben ser de 0 a 100.

Observabilidad, deberiamos tener metricas, tracing, controlado por los Ids.

Debemos asumir una tecnologia neutral a nivel de arquitectura, primero diseñemos la arquitectura, podemos hacer referencia a algun stack cuando presentemos el proyecto, pero por el momento olvidemos el lenguaje, empecemos por modulo monoliticos o pot descomposicion de servicios, asumir que single region deployment, duture multi region is a tretch goal, BD multitenant, imaginate un colegio internacional.

Las transacciones deben ser consistentes, especialmente cuando se inscribe alumnos y cuando se les pone notas

las bases de datos deben ser relacionales para las entidades principales. quisas o no, deberemos analizar, modelos de solo lectura, o caching, si vemos necesario usar 2, 3 bases de datos, analicemos y propongamoslo en la arquitectura.

cuando el jwt va a al backend se hace una consulta para verificar que haya un usuaqrio al que le corresponde estas credenciales, cuando eso pasa generamos un token que debe estar encriptado este token se lo enviamos de vuelta al frontend esta cookie se almacena en el frontend, a partir de que el usuario tiene acceso al sistema esta cookie debe ser validad siempre, en todos los requests se debe enviar el bearer. Cuando se realiza una request, se tiene cabecera y cuerpo, lo que se tiene en la cabecera son un chingo de cosas, en la cabcecera, se tiene siempre el bearer, que es la porcion del token que hemos generado, todos los tokens que tengan seguridad a traves de jwt deben llevar el valor en la cabecera, y en una bd debemos almacenar el identificador del usuario y el token que se le ha asignado para saber si es un token valido o no valido, a nivel backend debemos tener un middle weare y en ese middleware validamos ese token, el middleware es un servicio que se ejecuta antes de que el request llegue al conrtolador, es el lugar en el que se valida, CSRF, XSS, si se valida un token que no existe o que no lo hemos generado nosotros, no se le da acceso al controlador o nada que este detras del middleware. Cuando llegamos al backend, tenemos que buscar la manera de obtener este token, desencriptarlo y obtener el id del usuario. AAA, autenticacion autorizacion, accountability, solamente los usuarios permitidos pueden entrar al backend, autorizacion, solo los estudiantes pueden hacer x cosa, solo los profesores pueden hacer y cosa, accountability, se debe hacer al usuario responsable de todas las acciones que cometa. Quien ha cambiado la nota, de que estudiante, dia fecha, hroa, en que curso, en que hora, etc.

El estudiante debe poder ver sus notas, podríamos usar caching, vistas operacionales, administración de variables, donde van a estar almacenadas, como se las va a manejar, y un modelo de control de acceso justamente para la auditoria, roles y permisos, estos 3 roles.

analizar como vamos a medir la performance del sistema, resiliencia, que pasa cuando hay timeouts, habrá retries automaticos? manejaremos idempotencia, que haremos si dos profesores estan inngresado notas para un estudiante al mismo tiempo?