

1. Análisis de mejora

Si el sistema empieza a tener más usuarios al mismo tiempo, empezaría separando los módulos validar tamaños de conexión de la bd. Documentación de las funcionales y código fuente, manejo de buenas arquitecturas

2. Aplicación de buenas prácticas

- **¿Por qué es importante separar el código por responsabilidades?**
por el principio único de responsabilidad el cual nos dice que es importante separar las responsabilidades de cada clase también lo hace más escalable ya que un cambio en el código fuente no va afectar al momento de tener buenas prácticas.
- **¿Qué errores comunes ha cometido como desarrollador y cómo los ha corregido?**
Validaciones de campos en el fronted, los he solucionado mejorando las capacidades de análisis en los campos de formulario, en el backend mejor manejo de excepciones al momento y validaciones de datos
- **¿Qué herramientas usa para probar y depurar su código?**
Uso herramientas como el depurador del IDE (por ejemplo, VS Code), VisualStudio .net, BlobStorage para para ver errores y también hago pruebas con Postman