Marcos Cabral – Eda 3 Programación Web 3.

1. Como equipo hicimos una reunión donde repasamos los puntos que tenia que tener el trabajo practico. Seguido a esto, cada uno tomó la parte que mas le interesaba o llamaba la intención.

En mi caso debía realizar las funcionalidades de Gasto.

1. Creo que personalmente durante todo el trabajo practico suelen aparecer algunas complicaciones, como rutas de controladores que se vuelven inaccesibles y antes andaban, etc. Pero creo que en el que mas incapie hicimos como grupo fue en el procedimiento almacenado de expensas. Al principio lo habíamos resuelto con un procedimiento almacenado que joineaba las tablas y el calculo de expensas se resolvia en el repositorio, pero finalmente se “simplifico” todo en un solo cursor.

Mi desafio individual con el que tuve que lidiar fue el del archivocomprobante de Gasto, ya que era difícil de validarlo, cuando editabas podía desaparecer ya que el input aparecia como null, en resumen la validación y el agregado o quitado de un archivo al editar un gasto fue lo que más me complico.

El front del sitio también me dio algunas complicaciones, ya que había veces que visual no me reconocía los cambios en el css.

1. Una de las funcionalidades que podríamos mejorar seria el procedimiento almacenado, ya que podríamos ir joinneando cada objeto requerido de la base de datos, con consultas linq, y armar de una manera mas simple el cursor creado. De esta manera no es necesario que todo el equipo conozca cursores, ya que usamos de base el mismo lenguaje con el que se viene trabajando.
2. Con un filtro de sesión.
3. En líneas generales reutilizamos los métodos comunes a todos dentro del Repository y Service, ambos con un Base<>, tal como se vio en clases, de esta manera ahorramos mucho código para cada repo y servicio.

En mi caso donde mas reutilice código, fue con el ArchivoComprobante, ya que Cree una clase llamada FileVM que extiende de HttpPostedFileBase (el tipo de objeto que venia en el form cuando cargaba un pdf). El mismo tenia la funcionalidad de guardar un archivo en una carpeta, y la de manejar el las comprobaciones al momento de editar o cargar un archivo.

Esto es lo mencionado en el punto 2, donde quizás tengo un archivo en la bd y el file vino null, eso significa que edite pero no modifiqué el archivo. O si cargo un archivo y no tenía cargado uno antes (Alta). O finalmente, cuando cargo un archivo y tenía uno previamente (editar).

1. De mi parte, me encargue de incluir el procedimiento a visual e ir probando algunos procedimientos (antes del cursor), y el tipo de errores que había era por problemas de joins. Luego se solucionaron con el cursor, finalmente el error que teníamos era al momento de acceder a la ruta de la api, donde por un conflicto en el filtro nos re direccionaba a error 404. Lo más complejo fue el procedimiento almacenado.
2. Creo que con las correcciones marcadas por ustedes y corregidas por Fede, el trabajo debe tener un 95% de completitud.