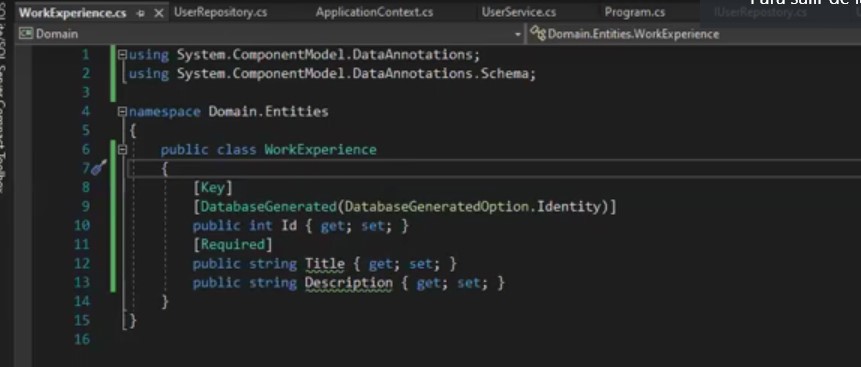
Patron repositorio

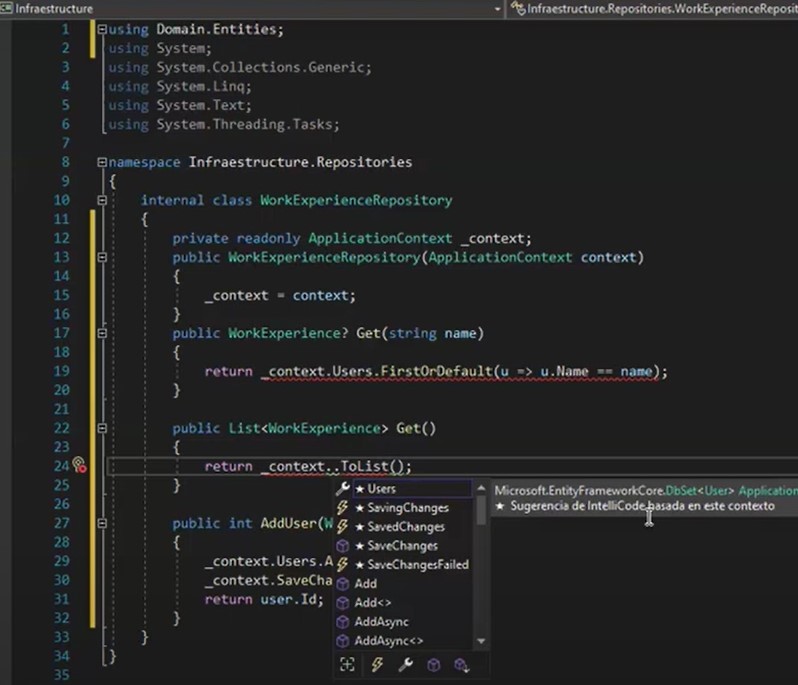


Key: decoradores para id

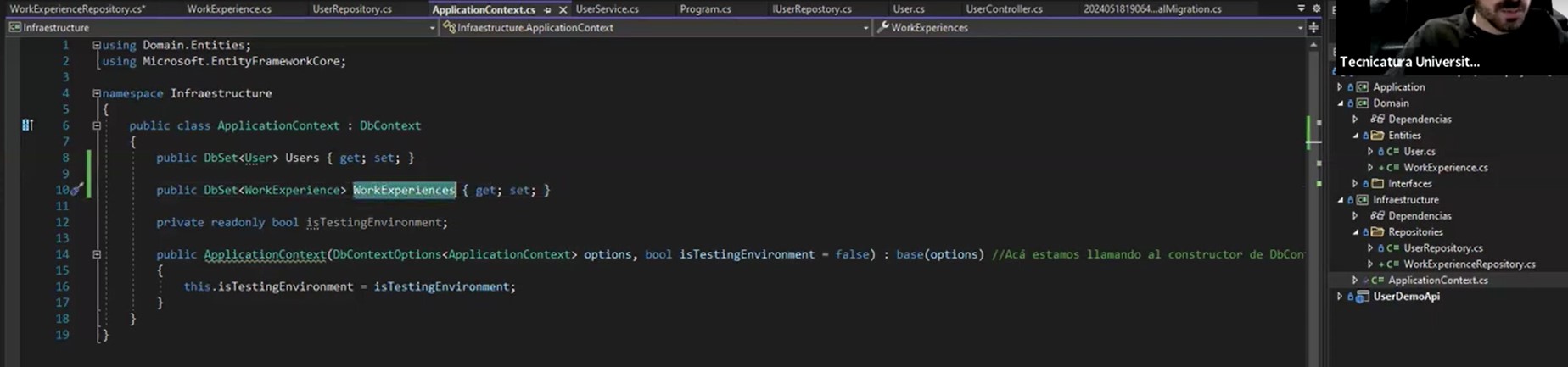
Databasegeneric: esto asigna un valor para la id en base a los que ya había en la db (eleva en 1 cada vez que pasa). Genera un id y lo devuelve cuando lo necesitamos

Hacemos el repositorio de work exprience

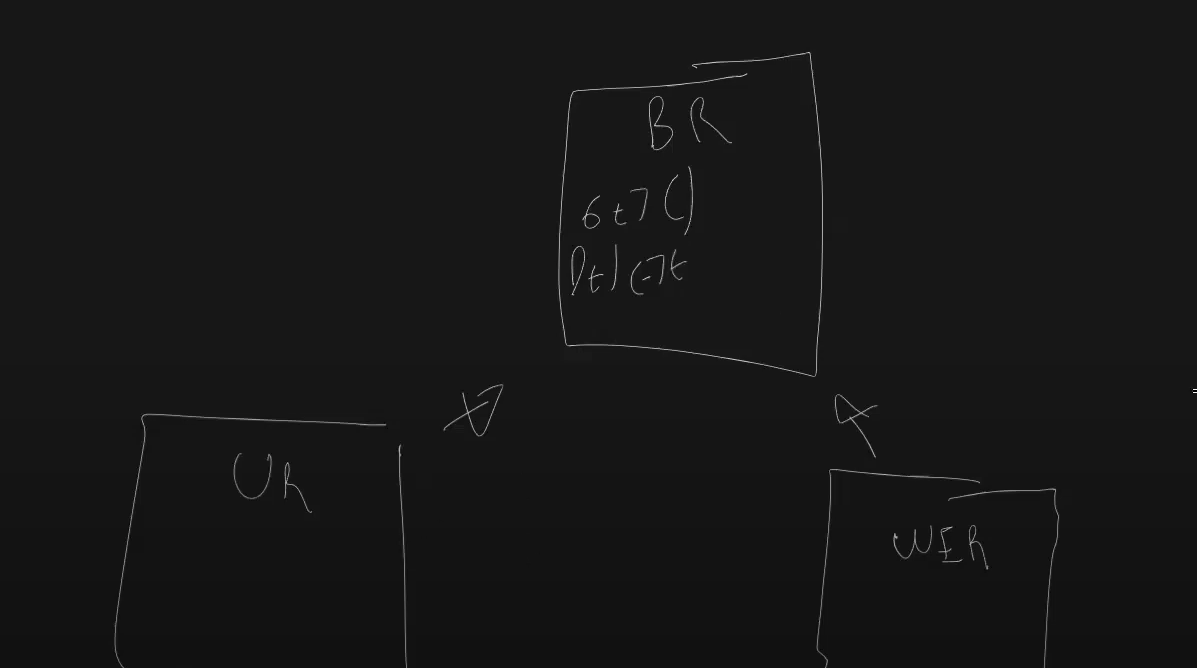
En este caso que tenemos que hacer para que tengamos disponible el conjunto de work experience y poder hacer linq?



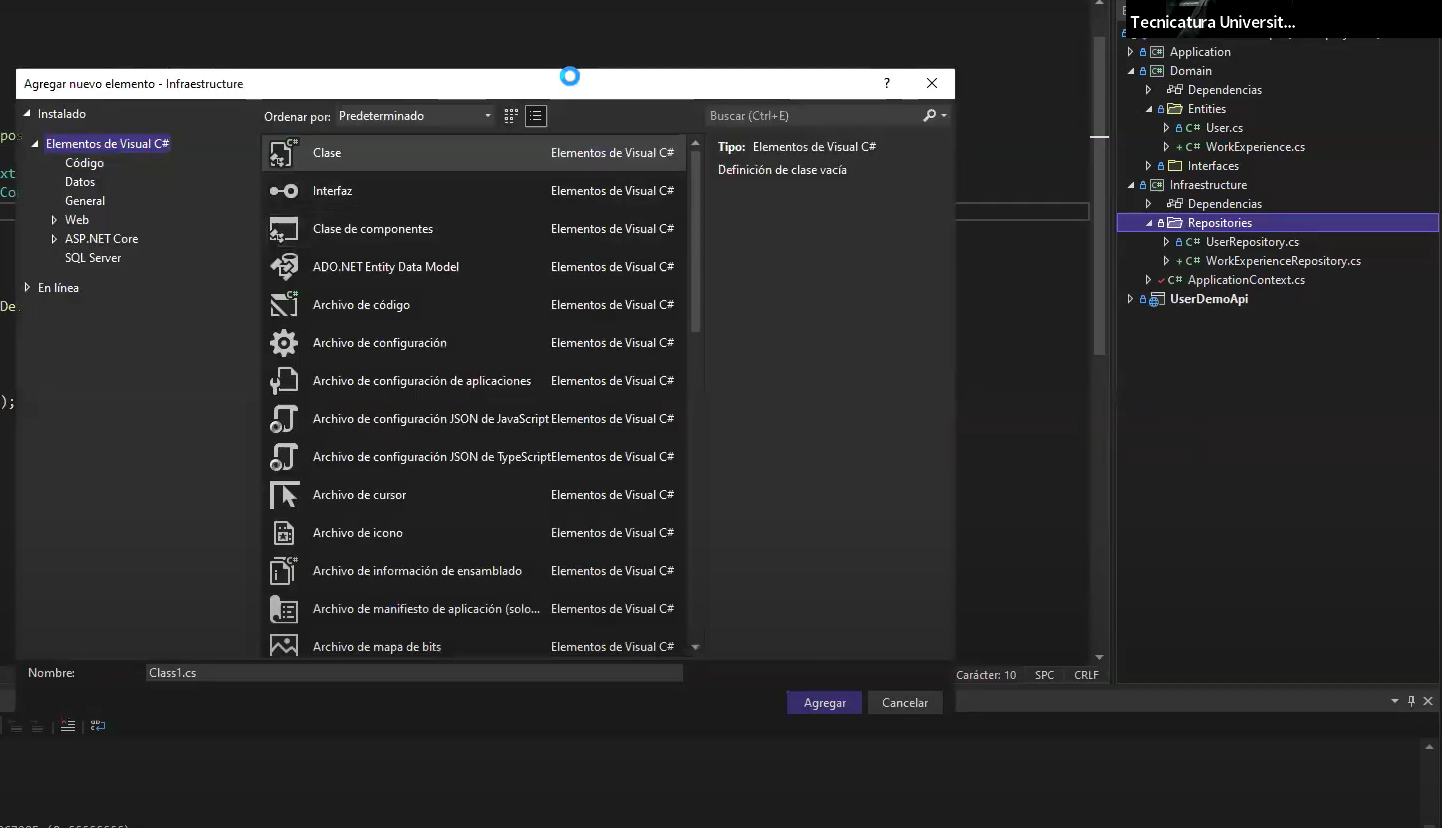
Tenes que ponerlo en context aplicación el DBSET



No se puede aplicar la herencia asi nomas pues los métodos de cada clase son distintos en userRepository por ejemplo el get te trae una lista de usuarios y el get de workexpirience devuelve una lista de work exprience entonces la herencia como tal no soluciona el problema de tener los métodos unificados en una clase, vamos a tener que seguir teniendo implementaciones distintas



Como se puede solucionar esto?

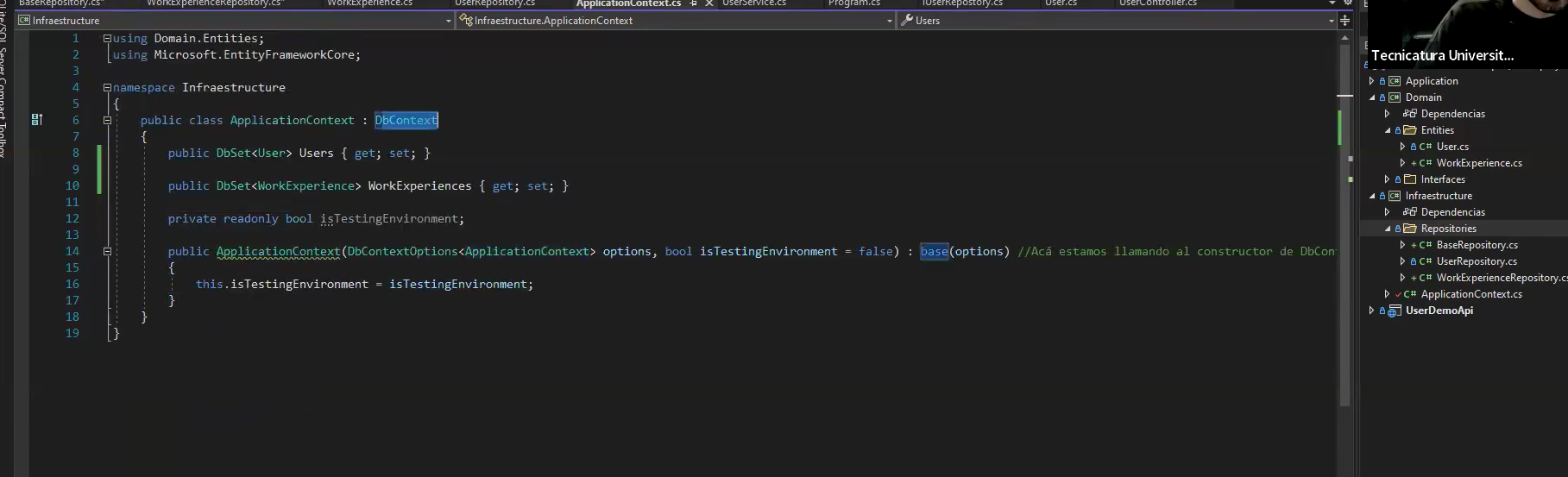


Creamos BaseRepository

Datos genéricos:

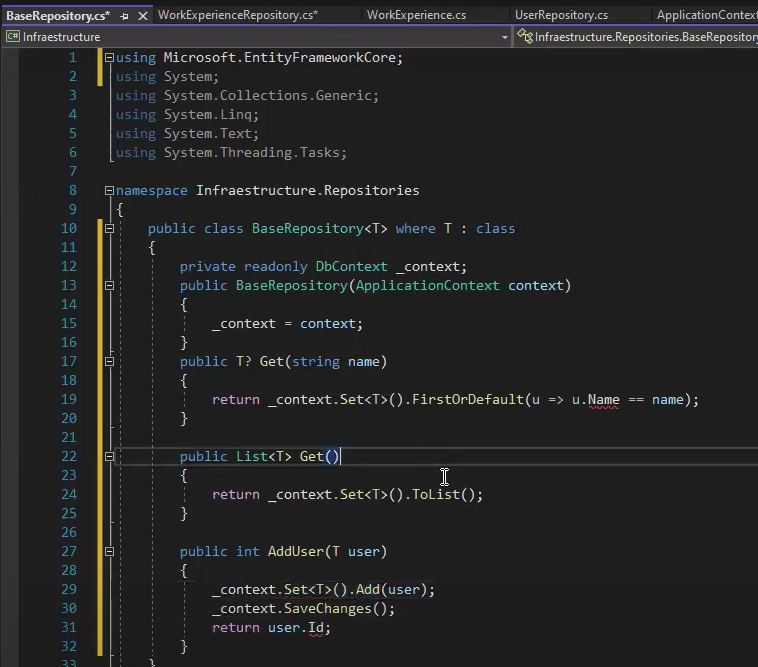
La clase como tal existe, la lista T existe, en este caso BaseRepository<T>, pero hasta que nadie la llama con lo que viene ahí dentro esa clase, no existe realmente.

Lo mismo pasa con las listas, si hay una lista y hasta que no pases lo que va ahí y definan la lista, la clase no existe

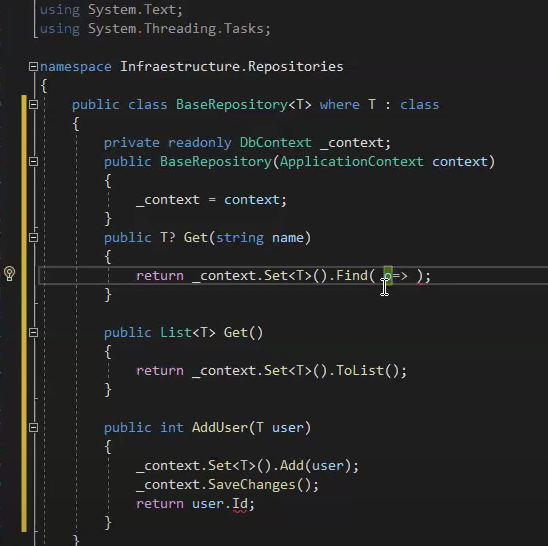


Db context es la clase q hereda

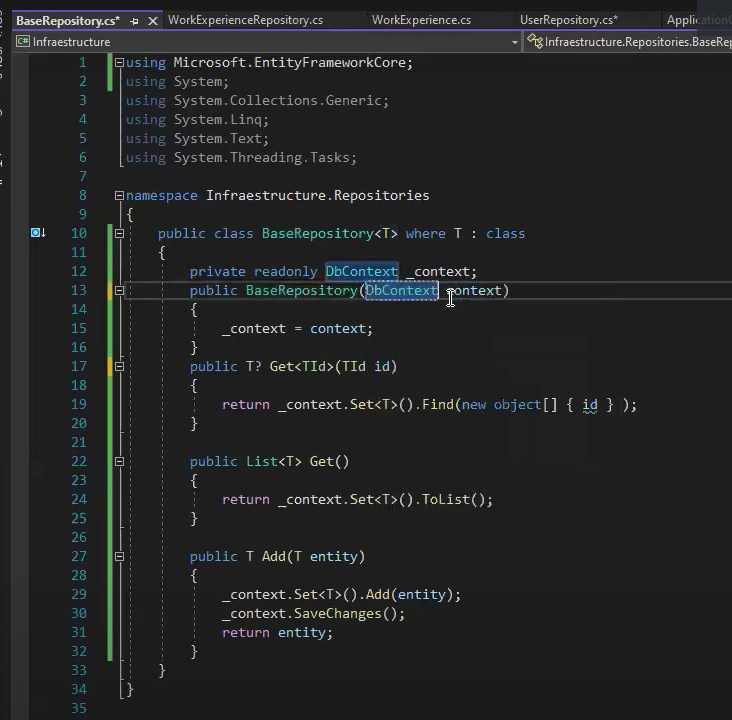
La necesitamos porque necesitamos db set y el dbSEt de T no existe, lo que si existe es entity framework que nos disponibiliza un método que es Set<T> que nos devuelve el dbset si existe uno.

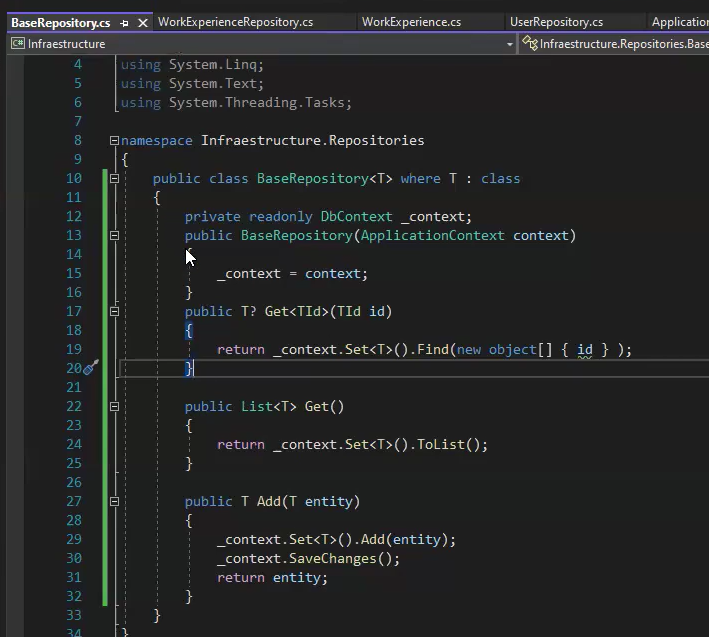


Con el set preguntas si existe un db de la entidad que me están pasando

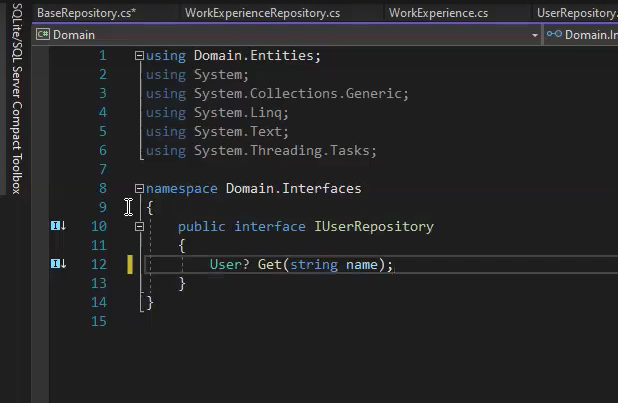


Tenes que usar un TI porque sino fallaría porque si pones id sola no te va a entender porque es T entonces por eso ponemos Tid que va a ser el identificador de T y el tipo de dato que pasamos no es un entero sino Tid

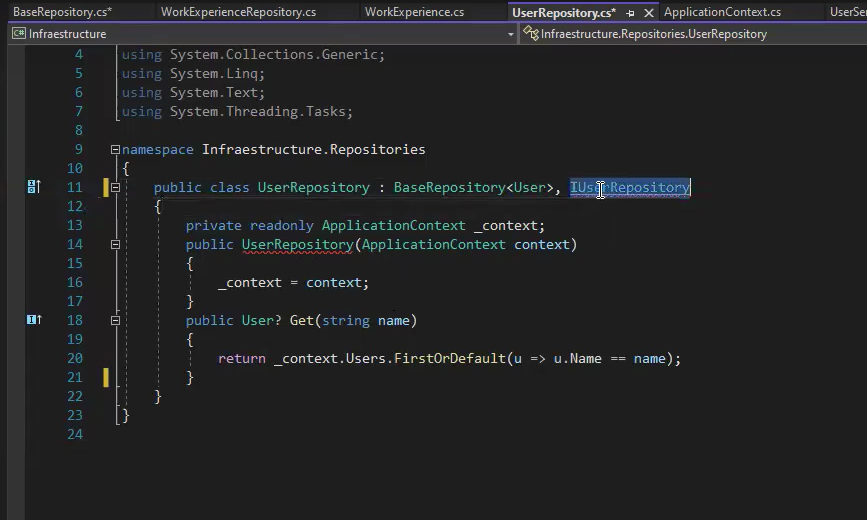




Ponemos que devuelva el objeto entero



Aca usamos base repostory user



Aquí debemos usar el base lo que haces es invocar la clase padre este base es palabra reservada, con esto que estas haciendo es invocar al constructor de la clase padre

Si vos tenes un método que tiene el nombre de la clase y paréntesis se llama constructor

