

Evidencia dia 01 - Semana 16

Leonardo Rodenas Escobar

Reflexión:

Las clases siguen tan "agradables" como siempre. Preguntar una duda me deja con 5 más. Intenté hacer unas correcciones de parte de un video de youtube y personalmente algo he comprendido, me falta hacer lo de los favoritos pero es mejor no preguntarlo en clases porque no hay respuesta o son desagradables.

Avance:

Muestro acá parte del avance del proyecto, para ver el avance completo revisar en el siguiente [link](#).

ViewModel:

```
class SismosViewModel(application: Application) : AndroidViewModel(application) {

    private var repositorio: SismosRepositorio

    init {

        val sismosDao =
            SismosDatabase.crearDatabase(application).obtenSismosDelDao()
        repositorio = SismosRepositorio(sismosDao)

    }

    fun traemeLoDelServer() {
        repositorio.obtenDataDelServer()
    }

    fun exponeSismosDeDB(): LiveData<List<SismosModel>> {
        return repositorio.exponeSismosDeLaBaseDeDatos()
    }

    fun exponeSismosFavoritosDeDB(): LiveData<List<SismosModel>> {
        return repositorio.exponeFavoritosDeLaBaseDeDatos()
    }

    fun cambiaeEstadoDeFavorito(sismo: SismosModel) {

        repositorio.cambiaeEstadoDeFavorito(sismo)

    }

    fun borraSismosFavoritosDeDB() {
        repositorio.borrarFavoritosDeLaBaseDeDatos()
    }

}
```

```
}
```

ViewModel Factory:

```
class Factory (private val application: Application) : ViewModelProvider.Factory {

    override fun <T : ViewModel> create(modelClass: Class<T>): T {
        if (modelClass.isAssignableFrom((SismosViewModel::class.java))) {
            return SismosViewModel(application) as T
        }
        throw IllegalArgumentException("ViewModel Desconocido")
    }

}
```

Database:

```
@Database(entities = [SismosModel::class], version = 1)
abstract class SismosDatabase: RoomDatabase() {

    abstract fun obtenSismosDelDao(): SismoDao

    companion object {

        @Volatile
        private var baseDeDatosCreada: SismosDatabase? = null

        fun crearDatabase(context: Context): SismosDatabase {

            if (baseDeDatosCreada == null) {
                synchronized(this) {
                    {
                        baseDeDatosCreada = Room.databaseBuilder(
                            context,
                            SismosDatabase::class.java,
                            "base_De_Datos_Sismos"
                        ).build()
                    }
                }
            }
            return baseDeDatosCreada!!
        }
    }

}
```



(función favorito implementado sólo a nivel de Toast, no como funcionalidad de la app aún)

Eso concluye mi avance por el día de hoy, muchas gracias.

Leonardo Rodenas Escobar