Ejercicio práctico final

El ejercicio de este problema es muy simple, pero te va a ayudar a recapitular varios conceptos.

Etapa 1

Descarga la aplicación del repositorio: https://github.com/eaceto/ticmas-lab-android(opens in a new tab)

Etapa 2

Abrí y corre la aplicación en tu emulador.

Etapa 3

Incorpora un nuevo test de UI que:

Verifique que el estado inicial del contador sea el correcto.

Haga 1 clic en el botón de decrementar.

Haga 1 clic en el botón de incrementar.

Verifique que el estado del contador sea con el valor 0.

2) En MainViewModelUnitTest cuando se corre el test salta error cuando no esta inicializado para corregir el error se tiene que crear una función:

@Before

fun setup () {

Dispatchers.setMain(dispacher)

viewModel = MainViewModel ()

}

Para que el dispatcher se resetee se tiene que crear la función:

@After

fun tearDown() {

Dispatchers.resetMain()

}

Una vez que corra el test pasa exitosamente.

Cuando vamos en la mainActivity.kt podemos interactuar con el contador incrementar o decrementar.

3) Para iniciar el contador y que sea 0 se tiene que crear la función:

@Test

fun mainViewModel\_CheckInitialValue() = runTest {

val value = viewModel.counter.value?.number

assertEquals(0, value)

}

Incrementar: Para incrementar se le agrega la función:

@Test

fun mainViewModel\_TestIncrementValue() = runTest {

launch {

viewModel.incrementCounter()

}

advanceUntilIdle()

val value = viewModel.counter.value?.number

assertEquals(1, value)

}

En el caso de que el usuario quiera presionar el botón dos veces se tiene que crear la siguiente función:

@Test

fun mainViewModel\_TestIncrementValueTwice() = runTest {

for (i in 0..1) {

launch {

viewModel.incrementCounter()

}

advanceUntilIdle()

}

val value = viewModel.counter.value?.number

assertEquals(2, value)

}

Decrementar: Para Decrementar se le agrega la función:

@Test

fun mainViewModel\_TestDecrementValue() = runTest {

launch {

viewModel.decrementCounter()

}

advanceUntilIdle()

val value = viewModel.counter.value?.number

assertEquals(-1, value)

}