

Universidad ORT Uruguay

Facultad de Ingeniería

Obligatorio Diseño Aplicaciones 2, TDD Justificacion

Luis Sempolis 185664

Entregado como requisito de la materia Diseño 2, Docente :
Gabriel Piffaretti

6 de mayo de 2021

Declaraciones de autoría

Nosotros, Luis Sempolis y Martin Vergara ; declaramos que el trabajo que se presenta en esa obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizábamos el primer obligatorio de Probabilidad y Estadística;
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad;
- Cuando hemos citado obras de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, hemos acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotros;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

Índice general

0.1.	Descripción de la estrategia de TDD	3
0.2.	Informe de las pruebas	3
0.2.1.	Prueba de Controllers	3
0.2.2.	Prueba de Lógica	3
0.2.3.	Prueba de Acceso A Datos	4
0.3.	Evidencia De Aplicación De TDD	4
0.3.1.	Ver las principales categorías y navegar entre su contenido y playlists, reproduciendo el contenido de forma pública.	4
0.3.2.	Mantenimiento de un psicólogo con su respectiva información	4
0.3.3.	Iniciar sesión en el sistema	5
0.3.4.	Agendar consultas con un psicólogo	5

0.1. Descripción de la estrategia de TDD

inside – out Siempre nos concentramos durante todo el trabajo en una entidad a la vez, Es decir cada entidad y la lógica asociada a la misma se creo una vez que tengamos las pruebas que satisfagan la funcionalidad, es decir que hagan necesaria la escritura de código.

Si bien podrían considerarse inútiles hasta que trabajen en conjunto .. esto nos permitió identificar que entidades y funcionalidades son independientes del resto del sistema.

0.2. Informe de las pruebas

0.2.1. Prueba de Controllers

Jerarquía	No cubiertos (bloques)	No cubiertos (% de bloques)	Cubiertos (bloques)	Cubiertos (% de bloques)
cocho_DESKTOP-MKNCJ7P 2021-05-06 18_56_13.coverage	341	15,05 %	1925	84,95 %
eldescontracturante.webapplication.dll	71	14,64 %	414	85,36 %
ElDescontracturante.WebApplication	43	100,00 %	0	0,00 %
WebApplication1.Controllers	3	1,20 %	248	98,80 %
AdministradorController	0	0,00 %	23	100,00 %
AudiosController	0	0,00 %	46	100,00 %
CategoriaController	2	4,08 %	47	95,92 %
LoginController	1	5,00 %	19	95,00 %
PlayListsController	0	0,00 %	35	100,00 %
PsicologoController	0	0,00 %	52	100,00 %
SolicitudController	0	0,00 %	26	100,00 %
WebApplication1.Models	25	13,09 %	166	86,91 %

Como vemos tenemos una cobertura solo un 1.20 de porcentaje no cubierto en los controllers.

No probamos los models por considerar estos no tener una lógica propensa a fallar. Con el uso de Mocks y la correctos pasos en una prueba hacemos que nuestras pruebas sean completas, verificables y atomizadas. Utilizamos en los Asserts los códigos de error http para probar que ante cualquier excepción inesperada sea realmente el código devuelto el correcto. El porcentaje obtenido nos garantiza una alta calidad en los controllers.

0.2.2. Prueba de Lógica

eldescontracturante.logicadominio.dll	175	63,64 %	100	36,36 %
ElDescontracturante.LogicaDominio	175	63,64 %	100	36,36 %
LogicaAdministrador	4	16,67 %	20	83,33 %
LogicaAudio	15	60,00 %	10	40,00 %
LogicaCategoria	0	0,00 %	14	100,00 %
LogicaCita	117	100,00 %	0	0,00 %
LogicaLogin	3	15,00 %	17	85,00 %
LogicaPlaylist	11	42,31 %	15	57,69 %
LogicaPsicologo	25	51,02 %	24	48,98 %
eldescontracturante.logicadominiotest.dll	0	0,00 %	255	100,00 %

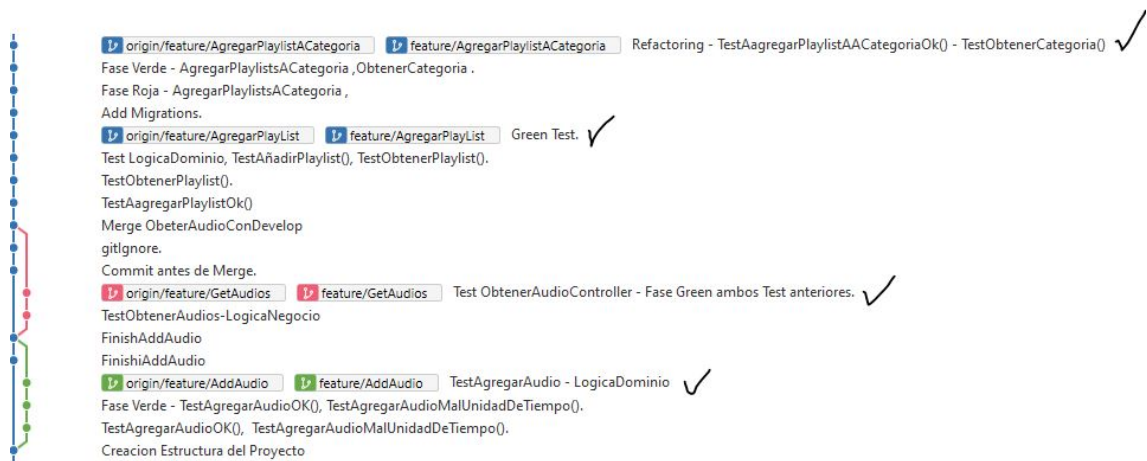
Mucha Lógica del Sistema se tuvo que hacer muy rápido por el riesgo de no poder entregar a tiempo, para esta la cobertura no fue la esperada ya que el proceso seguido de TDD no fue el mejor, es decir se escribió código sin tener una prueba que antes lo pruebe.

0.2.3. Prueba de Acceso A Datos

No se Realizo por no saber como usar base de datos en memoria.

0.3. Evidencia De Aplicación De TDD

0.3.1. Ver las principales categorías y navegar entre su contenido y playlists, reproduciendo el contenido de forma pública.



Como vemos siempre se paso a una nueva Feature una vez tenemos el test en Green de la Feature, tal vez nos falto un poco mas de etapas de refactoring para cumplir un TDD pulcro.

0.3.2. Mantenimiento de un psicólogo con su respectiva información



0.3.3. Iniciar sesión en el sistema



0.3.4. Agendar consultas con un psicólogo



Incluimos en este documento las funcionalidades que nos parecían claves.

Bibliografía

- [1] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, eight ed. McGraw-Hill, 2014.
- [2] I. Sommerville, *Software Engineering*, 10th ed. Pearson, 2015.