|  |  |
| --- | --- |
| UNMdP – F.I. – Ingeniería en Informática  Taller - I – TP Grupal 2024  Pautas para la entrega del T P Grupal 2024 | https://lh7-rt.googleusercontent.com/docsz/AD_4nXfmJvR0JtdZm2rLWUQVyAQNFrCONlv3V30d06wkh-cbQCTYE28YHUO03xvcCqijqtjPVK5ooI4L8tL1ygjDqyCRIi6VcaLR5WL-nQcnLsh_-LjMpsKE8DHx5N5m9_XZroB8Ix0rOR2a4YH_DXNQpVJsNyxJ?key=CfS6Z6nXt8m_lkdio5-YHA |

Este documento podrá ser modificado, debe revisar el enlace en el campus para ver las modificaciones

**Fechas de entrega y defensa**

Primer fecha de entrega obligatoria: Miércoles 13/11

Fecha de entrega del recuperatorio del TP: Miércoles 27/11 (solamente está disponible para quienes entregaron en la primer fecha y se les indicó que corrijan)

**Consideraciones**

1. Entregar un Proyecto (con maven) funcionando con el código fuente del testing y un Informe (pdf) que contenga al menos:
2. Carátula con descripción del grupo
3. Determinación datos de prueba y casos de pruebas unitarias de caja negra y de caja blanca (luego de realizar el test de cobertura se deberá realizar el test de caja blanca en aquellos métodos que presenten líneas de código sin ejecutar).
4. Determinación de casos de uso y casos de prueba para test de integración (orientado a objetos). Seguramente, esto va a arrojar casos de prueba ya contemplados en los test de unidad, no se debe repetir. Ver los pdf de la última clase, donde se habló del tema
5. Test de Persitencia. La persitencia consistirá en probar la escritura y lectura del archivo persistido cuando existe el archivo y cuando no. Pueden persistir todo o elegir alguna parte de los datos, esto lo debe justificar.
6. Test de GUI para las interfaces de usuario.
7. Testeo de excepciones también. Si las lanza, si las captura, si no las lanza, etc.
8. Adjuntar también el archivo persistido, así al probar el trabajo no debemos ingresar todos los datos necesariamente.
9. El informe debe incluir también un apartado con sus conclusiones, donde resaltarán los aspectos que consideren oportunos (nivel de dificultad de la experiencia, aprendizajes alcanzados durante el desarrollo del trabajo, etc.)
10. Para el día de la defensa del trabajo acompañar la presentación con un PPT.
11. El código debe subirse a GitHub, con toda la documentación dentro. En el campus debe subirse un texto que contenga:
12. número de grupo
13. integrantes
14. url del repositorio github
15. el nombre del repositorio debe contener el siguiente formato: 2024\_grupo\_X, donde X es el numero de grupo.

**Recomendaciones:**

1. recordar testear solamente lo permitido por los contratos, no se debe contradecir el contrato (no se podría establecer un valor esperado)
2. diferenciar bien los métodos de modelo, negocio y presentación (Vista + Control).
3. recordar que el test de GUI no tienen nada que ver con el Test de integración. Son dos cosas diferentes.
4. usar mocks donde fuera conveniente.

**Anexos**

**Prueba unitaria caja negra**

**Determinación de Datos de Prueba**

A partir de esta información se arman las pruebas de unidad. Esto se repite para cada escenario de cada clase de prueba.

Para cada constructor y método se utiliza la tabla:

**Tabla de Particiones en Clases de Equivalencia**

|  |  |  |  |
| --- | --- | --- | --- |
| Dato de entrada | Descripción de la Clases de equivalencia | Aplica ? (cumple el contrato?) | Identificador de clase de equivalencia |
| parámetro 1 | parámetro 1 > 0 | si | 1 |
| parámetro 1 | parámetro 1 < 0 | no | 2 |
| parámetro 2 | parámetro 2 contiene letras | si | 3 |
| … | … | … | … |

**Batería de Pruebas**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| número de prueba | Datos de entrada | Valor | Salida Esperada | clases de equiv. que abarca |
| 1 | parámetro 1 | 25 | especificar valor de retorno y valores de los atributos | 1 |
| parámetro 2 | “abc” | 3 |
|  | … | … | … | … |

Tener en cuenta lo siguiente:

1. Elección de la clase a testear, por ejemplo Biblioteca
2. Determinación de los escenarios, por ejemplo:
3. sin socios
4. con socios
5. Considerando cada escenario realizar la tabla de particiones y batería de pruebas para cada uno de los métodos y constructores que se necesita testear.
6. Es conveniente copiar en cada tabla de particiones el contrato del método/constructor y el de la clase, esto facilita determinar sí la clase de equivalencia aplica o no.