**Implementatietemplate Code Smell – Shothun Surgery Code Smell**

## Stubs & Mocks

|  |  |
| --- | --- |
| **Stub**  **1** | Plak hieronder de code van stub 1 (Stubclass) en de code die de Stub aanroept: |
| package example.org.stubs;  import example.org.logic.FunnyHintProvider;  public class StubFunnyHintProvider extends FunnyHintProvider {  @Override  public String geefHint(){  return "Dit is een funny stubhint!";  } } |
| Leg hieronder kort uit waarom jullie Stub 1 hebben gemaakt: |
| We hebben een stub gemaakt om te kijken of er daadwerkelijk een funny hint wordt aangegeven aan de speler. |
| **Stub**  **1** | Plak hieronder de code van stub 2 (Stubclass) en de code die de Stub aanroept: |
| package example.org.stubs;  import example.org.logic.HelpHintProvider;  public class StubHelpHintProvider extends HelpHintProvider {  @Override  public String geefHint(){  return "Dit is een stubhint!";  } } |
| Leg hieronder kort uit waarom jullie Stub 2 hebben gemaakt: |
| We hebben een stub gemaakt om te kijken of er daadwerkelijk een help hint wordt aangegeven aan de speler. |
| **Mock**  **1** | Plak hieronder de code van Mock 1 (Mockclass) en de code die de Mock gebruikt: |
| public class MockHintProvider {  @Test  public void testMonsterGeeftHintAlsAntwoordFoutIs() {  // Arrange  HintFactory mockFactory = *mock*(HintFactory.class);  HintProvider mockHint = *mock*(HintProvider.class);  *when*(mockFactory.getHint(*anyString*())).thenReturn(mockHint);  *when*(mockHint.geefHint()).thenReturn("Dit is een testhint!");   Speler mockSpeler = *mock*(Speler.class);  SpelerInventory mockInventory = *mock*(SpelerInventory.class);  *when*(mockSpeler.getInventory()).thenReturn(mockInventory);  *when*(mockInventory.gebruikItem(*anyString*(), *any*())).thenReturn(true);  *when*(mockSpeler.getHp()).thenReturn(100);    Monster monster = new Monster(50, mockSpeler, mockFactory);   // Act  monster.update(false, "ja", "potion", "ja");   // Assert  *verify*(mockInventory).gebruikItem("potion", monster);  *verify*(mockHint).geefHint();  } } |
| Leg hieronder kort uit waarom jullie Mock 1 hebben gemaakt: |
| In deze test gebruiken we mocks van Monster, SpelerInventory, Speler en HintProvider om te kijken of de onderlinge interacties correct verlopen. |
| **Mock**  **2** | Plak hieronder de code van Mock 2 (Mockclass) en de code die de Mock gebruikt: |
|  |
| Leg hieronder kort uit waarom jullie Mock 2 hebben gemaakt: |
|  |

**Bijdrage per student. Geef aan per student wat zijn of haar bijdrage was.**

|  |  |
| --- | --- |
| Naam student | Bijdrage |
| Dami Kastaneer | Stubs en Mock gemaakt. Tests gemaakt |
| Anson Mok | Code getest |
| Nathaniel Rasmijn | Code getest |
|  |  |