Swinburne University of Technology

COS20007 Object Oriented Programming

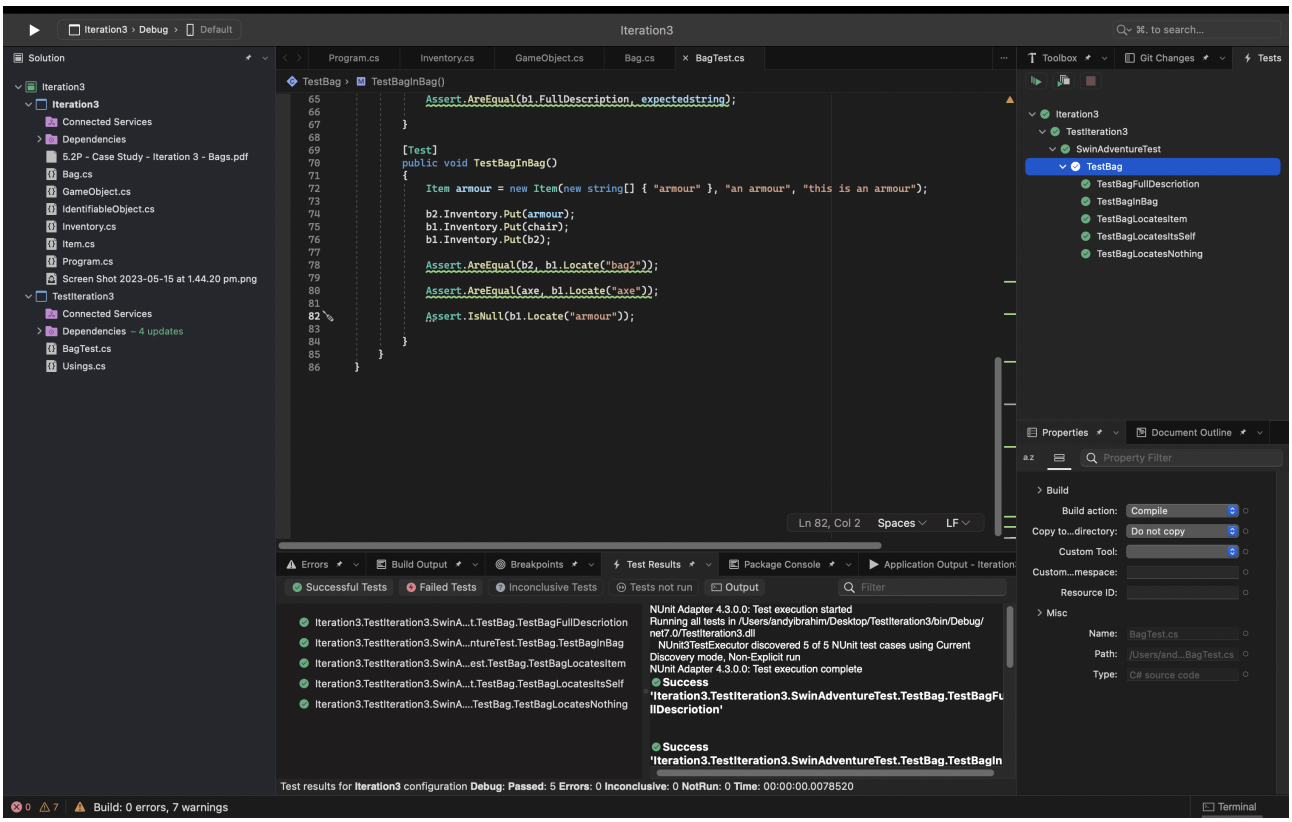
Case Study - Iteration 3 - Bags

PDF generated at 19:16 on Sunday 21st May, 2023

```
using System;

namespace SwinAdventure
{
    public class Bag : Item
    {
        private Inventory _inventory;

        public Bag(string[] ids, string name, string desc) : base(ids, name, desc)
        {
            _inventory = new Inventory();
        }

        public GameObject Locate(string id)
        {
            if (this.AreYou(id))
            {
                return this;
            }
            else
            {
                return _inventory.Fetch(id);
            }
        }

        public override string FullDescription
        {
            get { return $"in the {this.Name} you can see: " + _inventory.ItemList;}

        }

        public Inventory Inventory
        {
            get { return _inventory; }

        }
    }

}

```

```
using System;
using System.Numerics;
using System.Xml.Linq;
using NUnit.Framework;
using SwinAdventure;

namespace SwinAdventureTest
{
    [TestFixture]

    public class TestBag
    {

        Item axe;
        Item chair;
        Bag b1;
        Bag b2;


        [SetUp]
        public void SetUp()
        {
            axe = new Item(new string[] { "axe" }, "an axe", "this is an axe");
            chair = new Item(new string[] { "chair" }, "a chair", "this is a chair");
            b1 = new Bag(new string[] { "bag1" }, "a bag1", "This is a bag1");
            b2 = new Bag(new string[] { "bag2" }, "a bag2", "This is a bag2");
            b1.Inventory.Put(axe);

        }

        [Test]
        public void TestBagLocatesItem()
        {
            Assert.AreEqual(b1.Locate("axe"), axe);

            bool expected = true;
            bool actual = b1.Inventory.HasItem("axe");


            Assert.AreEqual(actual, expected);


        }

        [Test]
        public void TestBagLocatesItsSelf()
        {
            b1.Inventory.Put(b2);
            Assert.AreEqual(b2.Locate("bag2"), b2);


        }

```

```
54          [Test]
55          public void TestBagLocatesNothing()
56          {
57              Assert.IsNull(b1.Locate("synth"));
58          }
59
60
61          [Test]
62          public void TestBagFullDescriotion()
63          {
64              string expectedstring = "in the a bag1 you can see: an axe (axe)";
65              Assert.AreEqual(b1.FullDescription, expectedstring);
66
67          }
68
69          [Test]
70          public void TestBagInBag()
71          {
72              Item armour = new Item(new string[] { "armour" }, "an armour", "this is
    ↪  an armour");
73
74              b2.Inventory.Put(armour);
75              b1.Inventory.Put(chair);
76              b1.Inventory.Put(b2);
77
78              Assert.AreEqual(b2, b1.Locate("bag2"));
79
80              Assert.AreEqual(axe, b1.Locate("axe"));
81
82              Assert.IsNull(b1.Locate("armour"));
83
84          }
85      }
86  }
```


Iteration3 › Debug › Default
Iteration3
Solution
Iteration3
Iteration3
Connected Services
Dependencies
5.2P - Case Study - Iteration 3 - Bags.pdf
Bag.cs
GameObject.cs
IdentifiableObject.cs
Inventory.cs
Item.cs
Program.cs
Screen Shot 2023-05-15 at 1.44.20 pm.png
TestIteration3
Connected Services
Dependencies – 4 updates
BagTest.cs
Usings.cs
Program.cs
Inventory.cs
GameObject.cs
Bag.cs
BagTest.cs
TestBag › TestBagInBag()
Assert.AreEqual(b1.FullDescription, expectedstring);
}
[Test]
public void TestBagInBag()
{
Item armour = new Item(new string[] { "armour" }, "an armour", "this is an armour");
b2.Inventory.Put(armour);
b1.Inventory.Put(chair);
b1.Inventory.Put(b2);
Assert.AreEqual(b2, b1.Locate("bag2"));
Assert.AreEqual(axe, b1.Locate("axe"));
Assert.IsNull(b1.Locate("armour"));
}
}
}
Ln 82, Col 2
Spaces
LF
Errors
Build Output
Breakpoints
Test Results
Package Console
Application Output - Iteration
Successful Tests
Failed Tests
Inconclusive Tests
Tests not run
Output
Filter
Iteration3.TestIteration3.SwinA...t.TestBag.TestBagFullDescriotion
Iteration3.TestIteration3.SwinA...ntureTest.TestBag.TestBagInBag
Iteration3.TestIteration3.SwinA...est.TestBag.TestBagLocatesItem
Iteration3.TestIteration3.SwinA...t.TestBag.TestBagLocatesItsSelf
Iteration3.TestIteration3.SwinA....TestBag.TestBagLocatesNothing
NUnit Adapter 4.3.0.0: Test execution started
Running all tests in /Users/andybrahim/Desktop/TestIteration3/bin/Debug/net7.0/TestIteration3.dll
NUnit3TestExecutor discovered 5 of 5 NUnit test cases using Current Discovery mode, Non-Explicit run
NUnit Adapter 4.3.0.0: Test execution complete
Success 'Iteration3.TestIteration3.SwinAdventureTest.TestBag.TestBagFullDescriotion'
Success 'Iteration3.TestIteration3.SwinAdventureTest.TestBag.TestBagIn
Test results for Iteration3 configuration Debug: Passed: 5 Errors: 0 Inconclusive: 0 NotRun: 0 Time: 00:00:00.0078520
Toolbox
Git Changes
Tests
Iteration3
TestIteration3
SwinAdventureTest
TestBag
TestBagFullDescriotion
TestBagInBag
TestBagLocatesItem
TestBagLocatesItsSelf
TestBagLocatesNothing
Properties
Document Outline
Property Filter
Build
Build action: Compile
Copy to...directory: Do not copy
Custom Tool:
Custom...mespace:
Resource ID:
Misc
Name: BagTest.cs
Path: /Users/and...BagTest.cs
Type: C# source code
0 7
Build: 0 errors, 7 warnings
Terminal