Uppgift 6 – Integrationstestning

Eftersom jag i uppgift 3 valde att designa och implementera enbart den första delen av användningsfallet dvs. när aktören loggar in, så kommer jag även här att begränsa mig till det flödet. Det som kommer testas nu är hur de båda klasserna fungerar integrerade med varandra, för att simulera en inloggning från början till slut.

Integrationstest

Testklass som kördes:

Testfall 1 – Logga in med rätt användaruppgifter

```
class TestLoginAndUserDetails
  {

    public static void TestFullLogin()
    {
      string username = "username1";
      string password = "password1";
      Boolean userDetails = UserDetails.UserDetail(username, password);

      if(userDetails == true)
      {
        Boolean login = Login.SuccessfulLogin();

        if (login == true)
        {
          Console.WriteLine("Inloggning lyckades");
        }
        else
        {
          Console.WriteLine("Inloggning misslyckades");
        }
      }
      else
      {
        Boolean login = Login.UnsuccessfulLogin();

        if (login == false)
        {
          Console.WriteLine("Inloggning misslyckades");
        }
        else
        {
          Console.WriteLine("Inloggning lyckades");
        }
      }
    }
```

Testfall 2 – Logga in med fel användaruppgifter

```
class TestLoginAndUserDetails
  {

    public static void TestFullLogin()
    {
      string username = "username2";
      string password = "password2";
      Boolean userDetails = UserDetails.UserDetail(username, password);

      if(userDetails == true)
      {
        Boolean login = Login.SuccessfulLogin();

        if (login == true)
        {
          Console.WriteLine("Inloggning lyckades");
        }
        else
        {
          Console.WriteLine("Inloggning misslyckades");
        }
      }
      else
      {
        Boolean login = Login.UnsuccessfulLogin();

        if (login == false)
        {
          Console.WriteLine("Inloggning misslyckades");
        }
        else
        {
          Console.WriteLine("Inloggning lyckades");
        }
      }
    }
  }
```

Testresultat

Testfall 1: ”Inloggningen lyckades” (true).
Testfall 2: ”Inloggningen misslyckades” (false).

Klasserna som testades tillsammans

```
// Denna klass representerar användarens inloggningsuppgifter.
// Uppgifterna valideras och returnerar antingen rätt eller fel.

public class UserDetails
  {
    public static bool UserDetail(string username, string password)
    {
      if (username == "username1")
      {
        return true;
      }
      else
      {
        return false;
      }

      if (password == "password1")
      {
        return true;
      }
      else
      {
        return false;
      }
    }
  }

// Denna klass representerar ett logga-in scenario.
// Antingen lyckas användaren logga in eller inte.
  public class Login
  {
    public static bool SuccessfulLogin()
    {
      return true;
    }

    public static bool UnsuccessfulLogin()
    {
      return false;
    }
  }
```

Beskrivning

Min första tanke var att jag borde konstruera en testklass som kombinerar de tester jag tidigare utfört till ett enda test, som då kör de båda klasserna från användningsfallet efter varandra. Jag försökte använda mig av mallen från föreläsningen när jag utformade denna testbeskrivning. Efter detta så började jag implementera min nya testklass.

Jag märkte att min klass ”UserDetails” behövde ha något som returneras för att min plan för hur det här skulle fungera kunde genomföras. Jag ändrade därför detta i den klassen, vilket innebar att de föregående testklasserna nu inte kommer fungera. Jag behövde även ändra namn och typ av metod i klassen”. Jag fick slutligen kommentera ut klassen ”TestUserDetails” för att kunna köra koden jag gjort i min nya testklass. Dessutom kommenterade jag även ut de inaktuella testerna i Testrunner så att bara interaktionstestet körs.

Jag valde på grund av tidsbrist att endast göra tester på scenariot där användaren antingen skriver in rätt användaruppgifter eller fel användaruppgifter. Jag separerar alltså inte på om användaren skrivit endast sitt användarnamn fel eller bara lösenordet fel.