

Elaborato di Ingegneria del Software

Claudia Manfredi
Mattia Pavlovic
Elena Tonini

Università degli Studi di Brescia

A.A. 2021/2022

2022-07-10

Ingegneria Del Software

Elaborato di Ingegneria del Software

Claudia Manfredi
Mattia Pavlovic
Elena Tonini

Università degli Studi di Brescia

A.A. 2021/2022

Sommario I

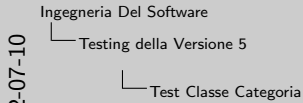
1. Testing della Versione 5	2
2. Pattern MVC	7
3. Pattern GRASP	7
4. Principi SOLID	7
5. Pattern della Gang of Four	7
6. Refactoring	7
7. Testing finale	7

Sommario I	
1. Testing della Versione 5	2
2. Pattern MVC	7
3. Pattern GRASP	7
4. Principi SOLID	7
5. Pattern della Gang of Four	7
6. Refactoring	7
7. Testing finale	7

Test Classe Categoria

Test di classe

```
class CategoriaTest {  
  
    12 usages  
    Nodo n = new Nodo( _nome: "Nodo", _descrizione: "categoria nodo");  
    5 usages  
    Foglia f = new Foglia( _nome: "Foglia", _descrizione: "categoria foglia");  
    2 usages  
    Foglia f1 = new Foglia( _nome: "Foglia", _descrizione: "categoria foglia");  
    1 usage  
    Foglia f2 = new Foglia( _nome: "Foglia", _descrizione: "categoria foglia");  
  
    @Test  
    void isNameTaken() {  
        n.addChild(f);  
        assertTrue(n.isNameTaken("Foglia"));  
    }  
  
    @Test  
    void nodeStructureIsValidIfHasNoChildren() { assertTrue(n.isStructureValid()); }
```



Per il test di classe della versione 5 abbiamo selezionato la classe Categoria in quanto permetteva di effettuare in modo efficiente test su numerosi metodi. Sarebbe diversamente stato complesso testare classi che richiedessero l'input da parte dell'utente.

Disclaimer: il codice della versione 5 era particolarmente disorganizzato e non rispettava alcun pattern. Con la versione finale post-refactoring è chiaramente più facile testare in modo più ordinato non solo questa classe ma anche tutte le altre.

I test effettuati hanno nomi esplicativi relativamente al fine del test stesso.

Test Classe Categoria

Ingegneria Del
Software

Claudia
Manfredi,
Mattia
Pavlovic,
Elena Tonini

Testing della
Versione 5

Pattern MVC

Pattern
GRASP

Principi
SOLID

Pattern della
Gang of Four

Refactoring

Testing finale

```
@Test
void nodeStructureIsValidIfHasOneChild(){
    n.addChild(f);
    assertFalse(n.isStructureValid());
}

@eleto
@Test
void nodeStructureIsValidIfHasTwoChildren(){
    n.addChild(f);
    n.addChild(f1);
    assertTrue(n.isStructureValid());
}

@eleto
@Test
void nodeStructureIsValidIfHasMoreThanTwoChildren(){
    n.addChild(f);
    n.addChild(f1);
    n.addChild(f2);
    assertTrue(n.isStructureValid());
}
```

2022-07-10

Ingegneria Del Software

Testing della Versione 5

Test Classe Categoria

Test Classe Categoria

```
@Test
void nodeStructureIsValidIfHasOneChild(){
    n.addChild(f);
    assertFalse(n.isStructureValid());
}

@eleto
@Test
void nodeStructureIsValidIfHasTwoChildren(){
    n.addChild(f);
    n.addChild(f1);
    assertTrue(n.isStructureValid());
}

@eleto
@Test
void nodeStructureIsValidIfHasMoreThanTwoChildren(){
    n.addChild(f);
    n.addChild(f1);
    n.addChild(f2);
    assertTrue(n.isStructureValid());
}
```

Test Classe Categoria

Ingegneria Del Software

Claudia Manfredi, Mattia Pavlovic, Elena Tonini

Testing della Versione 5

Pattern MVC

Pattern GRASP

Principi SOLID

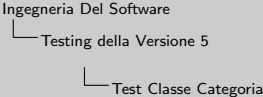
Pattern della Gang of Four

Refactoring

Testing finale

```
@Test
void leafStructureIsValid() { assertTrue(f.isStructureValid()); }
```

2022-07-10



Test Classe Categoria

```
@Test
void leafStructureIsValid() { assertTrue(f.isStructureValid()); }
```

Test Classe Orario

Test di requisiti e funzionalità

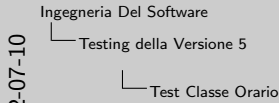
```
package tests;

import ...

1 usage  1 eletoo
class OrarioTest {
    3 usages
    Orario o1 = new Orario( hour: 10, minutes: 00);
    3 usages
    Orario o2 = new Orario( hour: 10, minutes: 10);
    3 usages
    Orario o3 = new Orario( hour: 10, minutes: 30);
    3 usages
    Orario o4 = new Orario( hour: 24, minutes: 00);
    3 usages
    Orario o5 = new Orario( hour: 00, minutes: 00);

    1 eletoo
    @Test
    void tenIsValidTime() { assertTrue(o1.isValid(o1.getHour(), o1.getMinutes())); }

    1 eletoo
    @Test
    void tenPastTenIsInvalidTime() { assertFalse(o2.isValid(o2.getHour(), o2.getMinutes())); }
```



Per il test delle funzionalità abbiamo selezionato la funzionalità:
“Gli orari per effettuare lo scambio possono essere solo allo scoccare dell’ora o alla mezza (ogni trenta minuti)”

Abbiamo applicato volutamente un testing “superficiale” sulla versione 5 in modo da poter mostrare l’evoluzione della tecnica utilizzata:

inizialmente i casi di test individuati erano in numero ridotto e poco approfonditi, mentre il testing della versione finale post-refactoring, come si vedrà nella sezione finale della presentazione, è stato effettuato in maniera più precisa e tenendo conto delle linee guida per l’individuazione dei casi di test black box.

Test Classe Orario

Ingegneria Del
Software

Claudia
Manfredi,
Mattia
Pavlovic,
Elena Tonini

Testing della
Versione 5

Pattern MVC

Pattern
GRASP

Principi
SOLID

Pattern della
Gang of Four

Refactoring

Testing finale

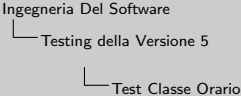
```

  @Test
  void tenThirtyIsValidTime() { assertTrue(o3.isValid(o3.getHour(), o3.getMinutes())); }

  @Test
  void midnightAtTwentyfourIsValidTime() { assertTrue(o4.isValid(o4.getHour(), o4.getMinutes())); }

  @Test
  void midnightIsValidTime() { assertTrue(o5.isValid(o5.getHour(), o5.getMinutes())); }
```

2022-07-10



Test Classe Orario

```

  @Test
  void tenThirtyIsValidTime() { assertTrue(o3.isValid(o3.getHour(), o3.getMinutes())); }

  @Test
  void midnightAtTwentyfourIsValidTime() { assertTrue(o4.isValid(o4.getHour(), o4.getMinutes())); }

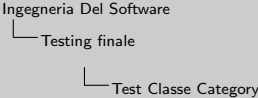
  @Test
  void midnightIsValidTime() { assertTrue(o5.isValid(o5.getHour(), o5.getMinutes())); }
```

- Ingegneria Del Software
- Claudia Manfredi, Mattia Pavlovic, Elena Tonini
- Testing della Versione 5
- Pattern MVC
- Pattern GRASP
- Principi SOLID
- Pattern della Gang of Four
- Refactoring
- Testing finale

Test Classe Category

Test di classe

2022-07-10



Test Classe Category
Test di classe

Abbiamo scelto di effettuare a seguito del refactoring gli stessi test effettuati sulla versione 5, in modo da verificare che le funzionalità corrette dell'applicazione rimanessero tali anche dopo il refactoring.

Test Classe Time

Test di requisiti e funzionalità

Ingegneria Del Software

Claudia Manfredi, Mattia Pavlovic, Elena Tonini

Testing della Versione 5

Pattern MVC

Pattern GRASP

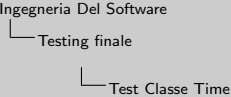
Principi SOLID

Pattern della Gang of Four

Refactoring

Testing finale

2022-07-10



Per il test delle funzionalità abbiamo mantenuto la scelta della funzionalità individuata per il testing iniziale:
“Gli orari per effettuare lo scambio possono essere solo allo scoccare dell’ora o alla mezza (ogni trenta minuti)”