Kamil Dusejovsky

Unit Testing Report

Table of Contents

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests

Kamil Dusejovsky

Task 2.1: Java unit tests

Original coverage before adding code.

Coverage jpacman [test]

Element	Class, %	Method, %	Line, %
nl.tudelft.jpacman	3% (2/55)	1% (5/312)	1% (14/1137)
board	20% (2/10)	9% (5/53)	9% (14/141)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)
game	0% (0/3)	0% (0/14)	0% (0/37)
integration	0% (0/1)	0% (0/4)	0% (0/6)
level	0% (0/13)	0% (0/78)	0% (0/345)
npc	0% (0/10)	0% (0/47)	0% (0/237)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	0% (0/6)	0% (0/45)	0% (0/119)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
LauncherSmokeTest	0% (0/1)	0% (0/4)	0% (0/29)
PacmanConfigurationExcepti	0% (0/1)	0% (0/2)	0% (0/4)

Coverage after adding my unit tests. Improvement in level, npc and sprite packages

Coverage jpacman [test]

Element	Class, %	Method, %	Line, %
nl.tudelft.jpacman	23% (13/55)	14% (46/312)	11% (136/1158)
board	20% (2/10)	9% (5/53)	9% (14/141)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)
game	0% (0/3)	0% (0/14)	0% (0/37)
integration	0% (0/1)	0% (0/4)	0% (0/6)
level	23% (3/13)	16% (13/78)	9% (32/351)
npc	40% (4/10)	12% (6/47)	6% (17/243)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	66% (4/6)	48% (22/45)	57% (73/128)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
LauncherSmokeTest	0% (0/1)	0% (0/4)	0% (0/29)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/4)

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests

Kamil Dusejovsky

I decided to work on these Unit Tests

Kamil Dusejovsky	src/main/java/nl/tudelft/jpacman/level/Player.setAlive
Kamil Dusejovsky	src/main/java/nl/tudelft/jpacman/level/Player.setKiller
Kamil Dusejovsky	src/main/java/nl/tudelft/jpacman/level/Player.addPoints
Kamil Dusejovsky	src/main/java/nl/tudelft/jpacman/level/Pellet.getValue
Kamil Dusejovsky	src/main/java/nl/tudelft/jpacman/level/Pellet.getSprite

Here is a code snippet of tests in package Pellet for methods getValue and getSprite

```
public class PelletTest {

    Sprite newSprite = new EmptySprite();
    Pellet newPellet = new Pellet(10, newSprite);


    @Test
    void testGetValue() {
        assertThat(newPellet.getValue()).isEqualTo(10);

    }

    @Test
    void testGetSprite() {
        assertThat(newPellet.getSprite()).isEqualTo(newSprite);

    }
}
```

and here is snippet for package Player method addPoints

```
@Test
void testAddPoints() {
    assertThat(object_player.getScore()).isEqualTo(0);
    object_player.addPoints(10);
    assertThat(object_player.getScore()).isEqualTo(10);
}
```

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests

Kamil Dusejovsky

Task 3: JaCoCo

1. JaCoCo coverage is slightly different from IntelliJ beause JaCoCo work on byteode level.

   IntelliJ interprets some code structures differently from JaCoCo

Here is an example of missed branch in getSprite() I missed the false branch of if statement

```
111.     }
112.
113.     @Override
114.     public Sprite getSprite() {
115.         if (isAlive()) {
116.             return sprites.get(getDirection());
117.         }
118.         return deathSprite;
119.     }
120.
121.     /**
```

Player

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
getSprite()		76%		50%	1	2	1	3	0	1
Player(Map, AnimatedSprite)		100%		n/a	0	1	0	7	0	1
setAlive(boolean)		100%		100%	0	3	0	7	0	1
addPoints(int)		100%		n/a	0	1	0	2	0	1
setKiller(Unit)		100%		n/a	0	1	0	2	0	1
isAlive()		100%		n/a	0	1	0	1	0	1
getKiller()		100%		n/a	0	1	0	1	0	1
getScore()		100%		n/a	0	1	0	1	0	1
Total	3 of 70	95%	1 of 6	83%	1	11	1	24	0	8

2. The source code visualization from JaCoCo is very helpful as example showed above.

3. I prefer JaCoCo because it makes it easier to see where my tests missed lines of code.

Task 4: Python

Here is the original test coverage.

```
Test Account Model
- Test creating multiple Accounts
- Test Account creation using known data

Name                    Stmts   Miss  Cover   Missing
-------------------------------------------------
models/__init__.py          7      0   100%
models/account.py          40     13    68%   26, 30, 34-35, 45-48, 52-54, 74-75
-------------------------------------------------
TOTAL                      47     13    72%
----------------------------------------------------------------------
Ran 2 tests in 0.180s

OK

(hw2) UNLV-Student@LON-2G07469 test_coverage %
```

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests

Kamil Dusejovsky

Here is the final test coverage.

```
(hw2) UNLV-Student@LON-2G07469 test_coverage % nosetests

Test Account Model
- Test creating multiple Accounts
- Test Account creation using known data
- Test account deletion
- Test account search
- Test dict to account
- Test the representation of an account
- Test account to dict
- Test account update

Name                 Stmts   Miss  Cover   Missing
--------------------------------------------------
models/__init__.py       7      0   100%
models/account.py       40      0   100%
--------------------------------------------------
TOTAL                   47      0   100%
----------------------------------------------------------------------
Ran 8 tests in 0.181s

OK

(hw2) UNLV-Student@LON-2G07469 test_coverage %
```

Code snippets:

```python
def test_from_dict(self):
    """ Test dict to account """
    data = ACCOUNT_DATA[self.rand]
    account = Account(**data)
    data = {
        'name': 'Spider man',
        'email': 'spider@man.com',
        'phone_number': '9996669990',
        'disabled': False,
        'date_joined': '2011-01-01'
    }
    account.from_dict(data)
    self.assertEqual(account.name, 'Spider man')
    self.assertEqual(account.email, 'spider@man.com')
    self.assertEqual(account.phone_number, '9996669990')
    self.assertEqual(account.disabled, False)
    self.assertEqual(account.date_joined, '2011-01-01')

def test_update(self):
    """ Test account update"""
    data = ACCOUNT_DATA[self.rand]
    account = Account(**data)
    account.id = 111111
    try:
        account.update()
    except DataValidationError:
        self.fail("DataValidationError in account update")
    account.id = None
    exception = False
    try:
        account.update()
    except DataValidationError:
        exception = True
    self.assertEqual(exception, True)

def test_delete(self):
    """ Test account deletion"""
    data = ACCOUNT_DATA[self.rand]
    account = Account(**data)
    account.create()
    account.delete()
    self.assertNotIn(account, ACCOUNT_DATA)
```

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests

Kamil Dusejovsky

```python
def test_find(self):
    """ Test account search"""
    data = ACCOUNT_DATA[self.rand]
    account = Account(**data)
    account.create()
    accountFound = Account.find(account.id)
    print(accountFound)
    print(account.id)
    self.assertEqual(account.name, accountFound.name)
```

Task 5: TDD

In this phase we focus on Test Driven Development which is separated by 3 phases.

Red Phase: Writing a failing test according to requirements

```
(hw2) UNLV-Student@LON-2G07469 tdd % nosetests

Counter tests
- It should create a counter
- It should return an error for duplicates (ERROR)

======================================================================
ERROR: It should return an error for duplicates
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/UNLV-Student/tdd/tests/test_counter.py", line 34, in test_duplicate_a_counter
    result = self.client.post('/counters/bar')
AttributeError: 'CounterTest' object has no attribute 'client'

Name             Stmts   Miss  Cover   Missing
---------------------------------------------
src/counter.py      11      1    91%   21
src/status.py        6      0   100%
---------------------------------------------
TOTAL               17      1    94%
----------------------------------------------------------------------
Ran 2 tests in 0.080s

FAILED (errors=1)
```

```
(hw2) UNLV-Student@LON-2G07469 tdd % nosetests

Counter tests
- It should create a counter
- It should return an error for duplicates
- read a counter
- It should update an existing counter (ERROR)

======================================================================
ERROR: It should update an existing counter
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/UNLV-Student/tdd/tests/test_counter.py", line 50, in test_update_a_counter
    self.assertEqual(data['fooRead'], baseline+1)
KeyError: 'fooRead'
-------------------- >> begin captured logging << --------------------
src.counter: INFO: Request to create counter: fooUpdate
-------------------- >> end captured logging << --------------------

Name             Stmts   Miss  Cover   Missing
---------------------------------------------
src/counter.py      22      2    91%   25, 32
src/status.py        6      0   100%
---------------------------------------------
TOTAL               28      2    93%
----------------------------------------------------------------------
Ran 4 tests in 0.063s

FAILED (errors=1)
```

Green Phase: Writing minimum amount of code to pass test.

```
(hw2) UNLV-Student@LON-2G07469 tdd % nosetests

Counter tests
- It should create a counter
- It should return an error for duplicates

Name             Stmts   Miss  Cover   Missing
---------------------------------------------
src/counter.py      13      1    92%   23
src/status.py        6      0   100%
---------------------------------------------
TOTAL               19      1    95%
----------------------------------------------------------------------
Ran 2 tests in 0.063s

OK
```

```
(hw2) UNLV-Student@LON-2G07469 tdd % nosetests

Counter tests
- It should create a counter
- It should return an error for duplicates
- read a counter
- It should update an existing counter

Name             Stmts   Miss  Cover   Missing
---------------------------------------------
src/counter.py      22      0   100%
src/status.py        6      0   100%
---------------------------------------------
TOTAL               28      0   100%
----------------------------------------------------------------------
Ran 4 tests in 0.063s

OK
```

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests

Kamil Dusejovsky

Refactoring: Cleaning up code with the assurance of our test

```
@app.route('/counters/<name>', methods=['POST'])
def create_counter(name):
    """Create a counter"""
    app.logger.info(f"Request to create counter: {name}")
    global COUNTERS
    if name in COUNTERS:
        return {"Message": f"Counter {name} already exists"},
status.HTTP_409_CONFLICT
    COUNTERS[name] = 0
    return {name: COUNTERS[name]}, status.HTTP_201_CREATED

@app.route('/counters/<name>', methods=['PUT'])
def update_counter(name):
    if name not in COUNTERS:
        return {"Message": f"Counter {name} doesnt exists"},
status.HTTP_404_NOT_FOUND
    COUNTERS[name] += 1
    return {name: COUNTERS[name]}, status.HTTP_200_OK

@app.route('/counters/<name>', methods=['GET'])
def read_counter(name):
    if name not in COUNTERS:
        return {"Message": f"Counter {name} doesnt exists"},
status.HTTP_404_NOT_FOUND
    return {name: COUNTERS[name]}, status.HTTP_200_OK
```

Exceptions and Errors Encountered with nosetests

- AssertionError: 404! =201
    - 404 http code NOT FOUND returned, but test expected 201 CREATED
- AttributeError: 'CounterTest' object has no attribute 'client'
    - During RED phase error occurred due to no code in counter.py
- AssertionError: 409! = 201
    - 409 CONFLICT http code returned but test expected 21 CREATED
- AttributeError: 'FlaskClient' object has no attribute 'update_counter'
    - During RED phase error occurred due to no code in counter.py

https://github.com/dusek2/UNLV-S24-CS472-Group7/tree/jpacman_tests