TDD Junit 5

Prática TDD - Jogo com *Player* e *Enemy*

André Luis Quiosi
Diego Corte Costa
Eduardo Richetti Bonatto

Player e *Enemy*

Para esse trabalho procurou-se desenvolver a ideia da criação de um jogo.

Usamos a ferramenta Junit 5 para realizar os casos de teste de Jogador (*Player*) e Inimigo (*Enemy*).

- Casos de teste Player
  - shouldAddExperience(): testa a adição de experiência na classe do jogador.
  - shouldLevelUp(): Testa se o player vai ou não subir de nível corretamente.
- Casos de teste Enemy
  - shouldSufferDamage(): Testa se o inimigo sofreu dano corretamente.
  - shouldDie(): Testa se o enemy morre quando a sua vida chega a zero.

Testes do *Player* – Classe PlayerTest

```java
public class PlayerTest {

    @Test
    public void shouldAddExperience() {
        Player player = new Player();
        int initialExperience = player.getExperience();

        int experience = 1000;
        player.addExperience(experience);

        int expectedExperience = initialExperience + experience;
        int newExperience = player.getExperience();

        assertEquals(newExperience, expectedExperience);
    }

    @Test
    public void shouldLevelUp() {
        Player player = new Player();
        int initialLevel = player.getLevel();

        int experience = 1000;
        int expectedUppedLevels = experience / Player.LEVEL_EXPERIENCE_QUANTITY;

        player.setExperience(experience);
        player.levelUp();

        int newLevel = player.getLevel();

        int uppedLevels = newLevel - initialLevel;

        assertEquals(uppedLevels, expectedUppedLevels);
    }
}
```

Testes do *Player* – Adicionar Experiência

```java
@Test
public void shouldAddExperience() {
    Player player = new Player();
    int initialExperience = player.getExperience();

    int experience = 1000;
    player.addExperience(experience);

    int expectedExperience = initialExperience + experience;
    int newExperience = player.getExperience();

    assertEquals(newExperience, expectedExperience);
}
```

Testes do *Player* – Subir de Nível

```java
@Test
public void shouldLevelUp() {
    Player player = new Player();
    int initialLevel = player.getLevel();

    int experience = 1000;
    int expectedUppedLevels = experience / Player.LEVEL_EXPERIENCE_QUANTITY;

    player.setExperience(experience);
    player.levelUp();

    int newLevel = player.getLevel();

    int uppedLevels = newLevel - initialLevel;

    assertEquals(uppedLevels, expectedUppedLevels);
}
```

Testes do *Enemy* – Classe EnemyTest

```java
public class EnemyTest {

    @Test
    public void shouldSufferDamage() {
        Enemy enemy = new Enemy(life:10);

        enemy.sufferDamage(damage:5);

        int expectedLife = 5;
        int enemyActualLife = enemy.getLife();

        assertEquals(enemyActualLife, expectedLife);
    }

    @Test
    public void shouldDie() {
        Enemy enemy = new Enemy(life:10);

        enemy.sufferDamage(damage:10);

        assertFalse(enemy.isAlive());
    }

}
```

Testes do *Enemy* – Recebimento de Dano

```java
@Test
public void shouldSufferDamage() {
    Enemy enemy = new Enemy(life:10);

    enemy.sufferDamage(damage:5);

    int expectedLife = 5;
    int enemyActualLife = enemy.getLife();

    assertEquals(enemyActualLife, expectedLife);
}
```

Testes do *Enemy* – Morte do Inimigo

```java
@Test
public void shouldDie() {
    Enemy enemy = new Enemy(life:10);

    enemy.sufferDamage(damage:10);

    assertFalse(enemy.isAlive());
}
```

Classe do Jogador – *Player*

```java
public class Player {
    public static final int LEVEL_EXPERIENCE_QUANTITY = 100;

    private int experience;
    private int level;

    public int getExperience() {
        return this.experience;
    }

    public int getLevel() {
        return this.level;
    }

    public void setExperience(int experience) {
        this.experience = experience;
    }

    public void setLevel(int level) {
        this.level = level;
    }

    public void addExperience(int experience) {
        setExperience(getExperience() + experience);
    }

    public void levelUp() {
        int levels = experience / Player.LEVEL_EXPERIENCE_QUANTITY;
        setLevel(levels);
    }
}
```

Classe do Jogador – *Player*

```java
public void addExperience(int experience) {
    setExperience(getExperience() + experience);
}

public void levelUp() {
    int levels = experience / Player.LEVEL_EXPERIENCE_QUANTITY;
    setLevel(levels);
}
```

Classe do Inimigo – *Enemy*

```java
public class Enemy {
    private int life;

    public Enemy(int life) {
        this.life = life;
    }

    public int getLife() {
        return life;
    }

    public void setLife(int life) {
        this.life = life;
    }

    public boolean isAlive() {
        return getLife() > 0;
    }

    public void sufferDamage(int damage) {
        setLife(getLife() - damage);
    }
}
```

Classe do Inimigo – *Enemy*

```java
public boolean isAlive() {
    return getLife() > 0;
}

public void sufferDamage(int damage) {
    setLife(getLife() - damage);
}
```