Test Driven Development (TDD)

Contents

Contents

1 Why would someone test code

1 Why would someone test code

- Guarantee that my code works correctly

Contents

2 Disadvantages

2 Disadvantages

- Takes a lot of time in the beginning

2 Disadvantages

- Takes a lot of time in the beginning
  - Set up testing tool :(

2 Disadvantages

- Takes a lot of time in the beginning
  - Set up testing tool :(
  - I don’t know how the testing framework works. I have to work into that :(

2 Disadvantages

- Takes a lot of time in the beginning
  - Set up testing tool :(
  - I don’t know how the testing framework works. I have to work into that :(
- I don’t produce features with testing.

2 Disadvantages

- Takes a lot of time in the beginning
  - Set up testing tool :(
  - I don’t know how the testing framework works. I have to work into that :(
- I don’t produce features with testing. My employer wants that feature tomorrow, so I am faster if I skip testing :P

2 Disadvantages

- Takes a lot of time in the beginning
  - Set up testing tool :(
  - I don’t know how the testing framework works. I have to work into that :(
- I don’t produce features with testing. My employer wants that feature tomorrow, so I am faster if I skip testing :P

Right?

Contents

1) Why would someone test code
2) Disadvantages
3) Advantages
4) How do I work test driven
5) Example

3 Advantages

3 Advantages

- Stable software

3 Advantages

- Stable software
- Guarantee that old code still works after an update

3 Advantages

- Stable software
- Guarantee that old code still works after an update
- Automatically run tests

3 Advantages

- Stable software
- Guarantee that old code still works after an update
- Automatically run tests
- Save time during middle and large projects

Contents

1) Why would someone test code
2) Disadvantages
3) Advantages
4) How do I work test driven
5) Example

4 How do I work test driven

4 How do I work test driven

- Write test code before the implementation [1]

[1] Source: Lech Madeyski, *Test-Driven Development An Empirical Evaluation of Agile Practice*, Springer, 2010, Page 1

4 How do I work test driven

- Write test code before the implementation [1]
  - Test will always fail first

[1] Source: Lech Madeyski, *Test-Driven Development An Empirical Evaluation of Agile Practice*, Springer, 2010, Page 1

4 How do I work test driven

- Write test code before the implementation [1]
  - Test will always fail first
- What to test?

[1] Source: Lech Madeyski, *Test-Driven Development An Empirical Evaluation of Agile Practice*, Springer, 2010, Page 1

4 How do I work test driven

- Write test code before the implementation [1]
  - Test will always fail first
- What to test?
  - Call your function with null, 0, -1, empty string, very big numbers, very small numbers

[1] Source: Lech Madeyski, *Test-Driven Development An Empirical Evaluation of Agile Practice*, Springer, 2010, Page 1

4 How do I work test driven

- Write test code before the implementation [1]
  - Test will always fail first
- What to test?
  - Call your function with null, 0, -1, empty string, very big numbers, very small numbers
  - Check ranges. Is last and first element accessible? Do you get access violation?

[1] Source: Lech Madeyski, *Test-Driven Development An Empirical Evaluation of Agile Practice*, Springer, 2010, Page 1

4 How do I work test driven

- Write test code before the implementation [1]
  - Test will always fail first
- What to test?
  - Call your function with null, 0, -1, empty string, very big numbers, very small numbers
  - Check ranges. Is last and first element accessible? Do you get access violation?
- Try to crash your application

[1] Source: Lech Madeyski, *Test-Driven Development An Empirical Evaluation of Agile Practice*, Springer, 2010, Page 1

Contents

1) Why would someone test code
2) Disadvantages
3) Advantages
4) How do I work test driven
5) Example

5 Example

5 Example

- `char charAt(string s, int index) {}`

5 Example

- `char charAt(string s, int index) {}`
- In a test you define what you expect from a function

5 Example

- `char charAt(string s, int index) {}`
- In a test you define what you expect from a function
- In the above example we expect that the function returns the character in string `s` which is at the position `index`.

5 Example

- `char charAt(string s, int index) {}`
- In a test you define what you expect from a function
- In the above example we expect that the function returns the character in string `s` which is at the position `index`.
- `assertEquals('o', charAt("world", 1));`

5 Example

- `char charAt(string s, int index) {}`
- In a test you define what you expect from a function
- In the above example we expect that the function returns the character in string `s` which is at the position `index`.
- `assertEquals('o', charAt("world", 1));`
- What happens if you pass -1 for `index` ?

5 Example 2

5 Example 2

- `float divide(int a, int b) {}`

5 Example 2

- `float divide(int a, int b) {}`
- The above function shall return a / b

5 Example 2

- `float divide(int a, int b) {}`
- The above function shall return a / b
- How to test:

5 Example 2

- `float divide(int a, int b) {}`
- The above function shall return a / b
- How to test:
- `assertEquals(1.2, divide(12, 10));`

5 Example 2

- `float divide(int a, int b) {}`
- The above function shall return a / b
- How to test:
- `assertEquals(1.2, divide(12, 10));`
- `assertEquals(0.0, divide(1, 0));`

5 Example 2

- `float divide(int a, int b) {}`
- The above function shall return a / b
- How to test:
- `assertEquals(1.2, divide(12, 10));`
- `assertEquals(0.0, divide(1, 0));`
- What happens in the second test `divide(1, 0)`?

Would you care about testing?

Would you care about testing?

- If you code for yourself

Would you care about testing?

- If you code for yourself
- If you code for friends

Would you care about testing?

- If you code for yourself
- If you code for friends
- For customers

Would you care about testing?

- If you code for yourself
- If you code for friends
- For customers
- In medical use