Unit Testing

Automated Testing

by Sunit & Sarthak

Why automated testing?

- Self Testing Code http://www.martinfowler.com/bliki/SelfTestingCode.html
- Automated Testing is a necessary part of Continuous Integration (CI)
- Help to achieve frequent releases a.k.a Continuous Delivery (CD)
- We can’t be agile without automated testing (unit testing)

Unit Tests

- AAA of Unit Testing
- Clear separation between Arrange, Act and Assert
- Give descriptive method name for each test
- One test one scenario
- Focus on Readability
-

```
@Test
public void longDescriptiveName() {

    // arrange  (Setup)

    // Arrange all necessary preconditions and inputs.

    // act      (Execute)

    // Act on the object or method under test.

    // asset  (Verify)

    // Assert that the expected results have occurred.

}
```

Key points for Unit Testing

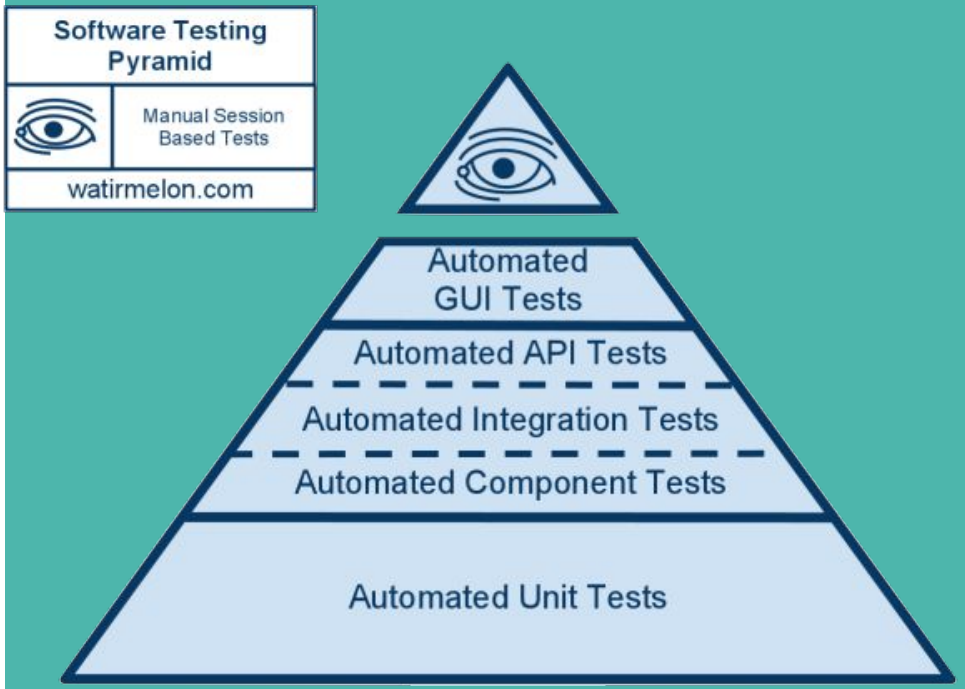
- Speed
- Repeatable
- Robustness
- Reliable
- Debugging

Test Pyramid

What kind of automated tests should we write?

What is the right level of testing?

How much of test coverage is “good enough” ?



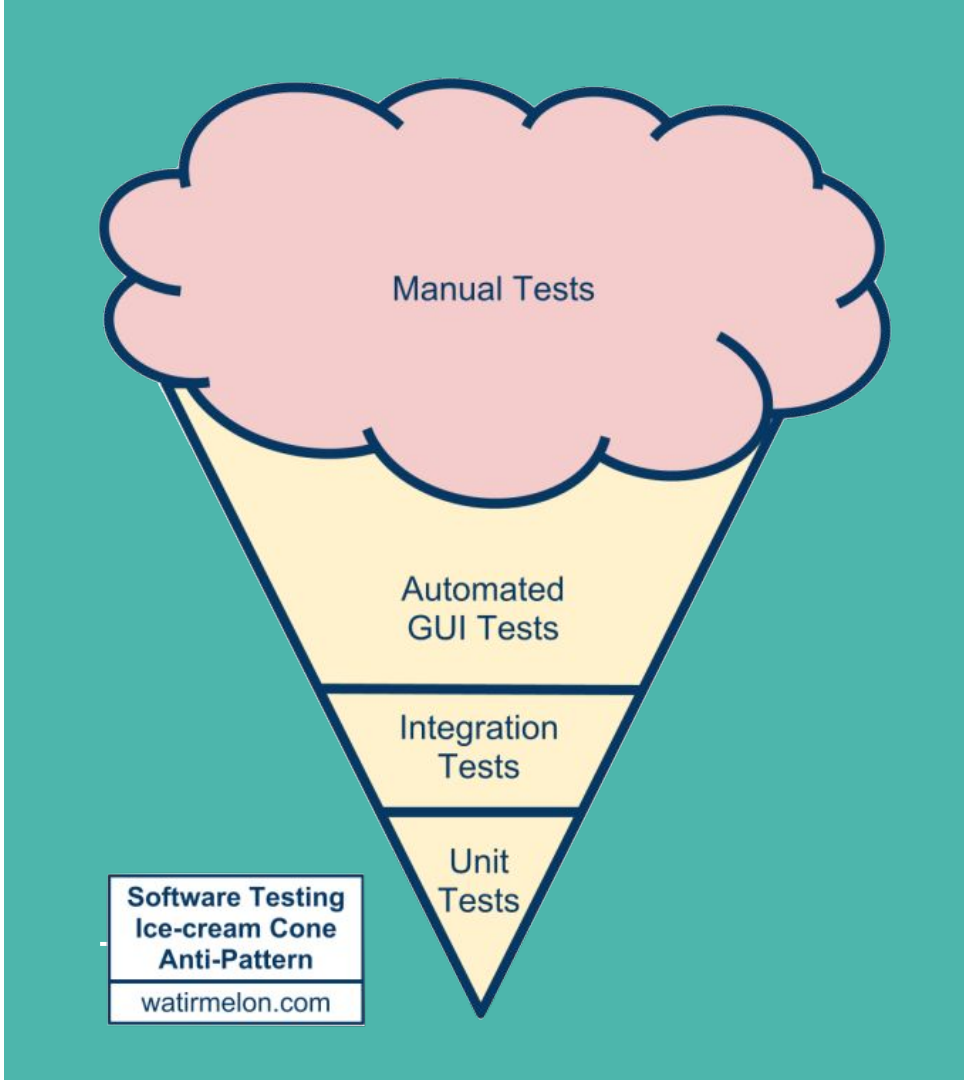
http://martinfowler.com/bliki/TestPyramid.html

http://www.thoughtworks.com/insights/blog/guidelines-structuring-automated-tests

Ice Cream Cone (Anti Pattern)

‘inverted’ pyramids of software testing looks like ‘ice-cream cone’

http://watirmelon.com/2012/01/31/introducing-the-software-testing-ice-cream-cone/



Test Driven Development (TDD)

- A journey from “Test Last” to “Test First”
- TDD to Influence Design
  http://www.thoughtworks.com/insights/blog/using-tdd-influence-design
-

Unit Testing Myths

- Need more time to write unit tests
- Unit Test code need not follow clean code practices
- Hitting database is not right in unit testing
- Separate team to write automated tests

Are you ready?

- Let’s take unit testing as implicit part of our learning and development
- Let’s avoid writing any code without unit testing.

Future Reading

- Use Builder Pattern to make Setup easy and readable
  - http://www.kenneth-truyers.net/2013/07/15/flexible-and-expressive-unit-tests-with-the-builder-pattern/
- Mocking http://martinfowler.com/articles/mocksArentStubs.html
- Mocking using Mockito http://gojko.net/2009/10/23/mockito-in-six-easy-examples/

Thanks!

Like to reach us,

Sunit Parekh
parekh.sunit@gmail.com

Sarthak Makhija
sarthak.makhija@gmail.com

Mocking

- When to Mock?
  - Isolate dependency for unit testing
  - External dependencies which are out of control of unit tests, e.g. web service calls
  - Time travel tests
- If possible, do mocking when necessary else runs into mocking hell