

# Software Testing Anti-patterns

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

# Software testing

[https://en.wikipedia.org/wiki/Software\\_testing](https://en.wikipedia.org/wiki/Software_testing)

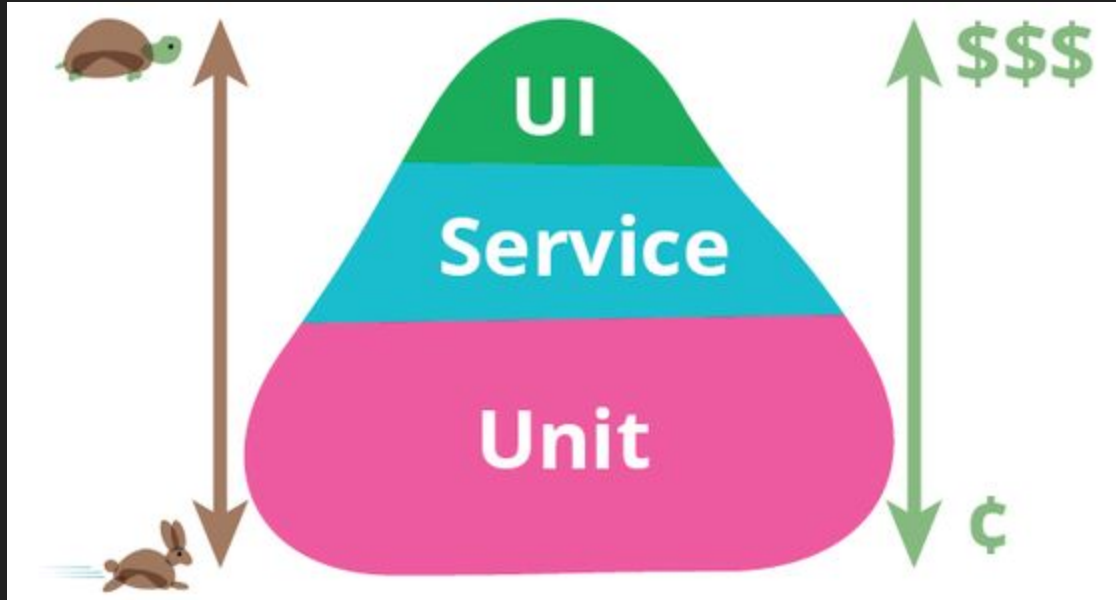
# Software design pattern

[https://en.wikipedia.org/wiki/Software\\_design\\_pattern](https://en.wikipedia.org/wiki/Software_design_pattern)

# Anti-pattern

<https://en.wikipedia.org/wiki/Anti-pattern>

# Test Automation Pyramid



<https://martinfowler.com/bliki/TestPyramid.html>

# Giving testing a bad reputation out of ignorance

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

# Running tests manually

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

Testing is only done by  
testers



# Flaky or slow tests

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

# Treating test code as a second class citizen

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

# Wrong kind of tests

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

# Testing the wrong functionality

<http://blog.codepipes.com/testing/software-testing-antipatterns.html>

# No test strategy

[https://en.wikipedia.org/wiki/Test\\_strategy](https://en.wikipedia.org/wiki/Test_strategy)