



eSolutions

ICT Volume 3 : Application Standards

ICT 3.3.1-2014 Web Application Testing Standards

Abstract

This document defines the standards applicable to web application testing.

All rights reserved. No part of this work covered by Deakin University's copyright may be reproduced or copied in any form or by any means (graphic, electronic or mechanical, including photocopying, recording, taping or information retrieval systems) without the written permission of Deakin University.

Document Control

Document Title	ICT 3.3.1-2014 Web Application Testing Standards
Version	2014
Controlled Copy (Electronic Reference)	

Document History

Ver.	Primary Author(s)	Description of Version	Date Completed
0.01	Steven George	Initial draft	26-11-2008
0.02	Steven George	Standards compliance	25-02-2009
0.03	Steven George	Standards compliance	18-03-2009
0.04	Steven George	Correct numbering error	07-04-2009
1.0	Steven George	Annual Review	05-07-2010
2.0	Steven George	2011 Update	02-11-2010
2.1	Michael Heley	Standards Compliance Review	11-09-2014

Table of Contents

1 UNIT TESTING 6

1.1 WEB APPLICATIONS SHOULD EMPLOY UNIT TESTING WHERE PRACTICAL 6

1.2 WHEN TO WRITE UNIT TESTS 6

1.3 TEST COVERAGE 6

1.4 TESTS MUST BE WRITTEN IN A WAY THAT SPECIFIC AND STRICT..... 6

1.5 TESTS MUST BE AUTOMATED 6

1.6 TESTS MUST BE ISOLATED 6

1.7 TESTS SHOULD TAKE A MICRO VIEW OF THE ARCHITECTURE 6

1.8 UNIT TESTS SHALL BE CREATED IN SUITES 6

1.9 TEST CLASS AND METHOD NAMING SHALL CONFORM WITH PHPUnit CONVENTIONS 7

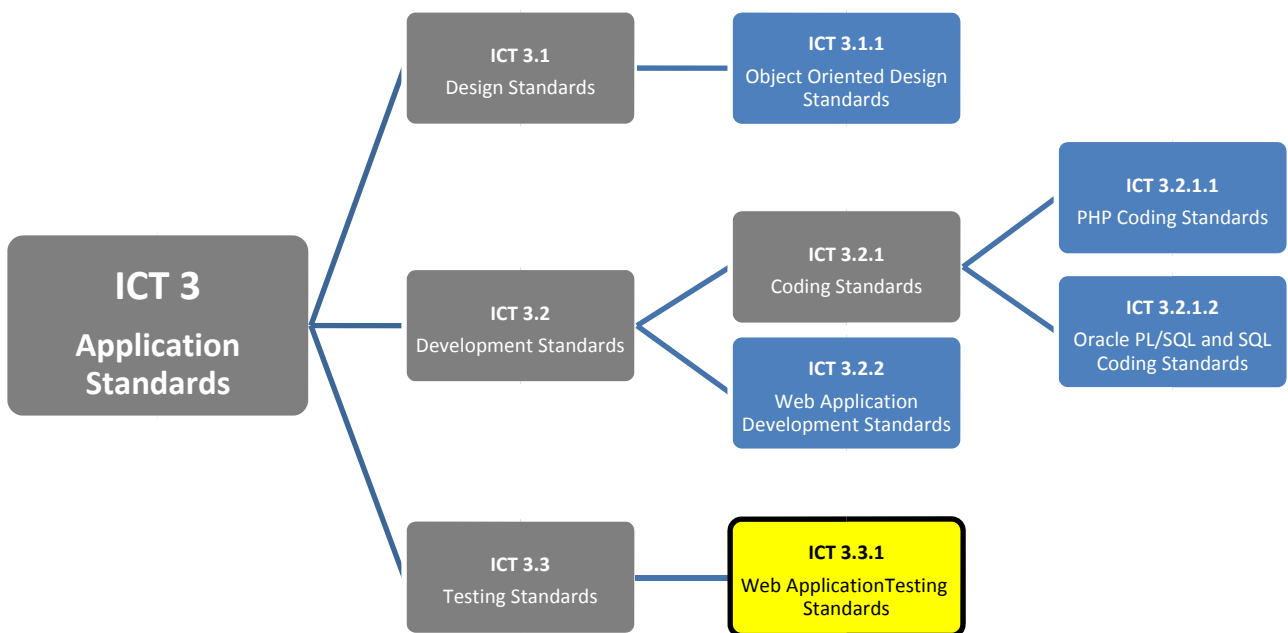
1.10 TESTS MUST BE KEPT UP TO DATE 7

1.11 UNIT TESTS ASSOCIATED WITH A SPECIFIC CLASS MUST BE RUN WHEN THE CLASS HAS BEEN CHECKED IN TO THE SOURCE CONTROL SYSTEM 7

2 SECURITY TESTING 7

2.1 THE OWASP TESTING GUIDE SHOULD BE USED..... 7

ICT Volume 3 : Application Standards



Standards Brief

This document serves to outline standards that shall apply within Deakin University.

Standard Document Access

All Deakin University staff and authorised/approved contracted personnel are provided access to this document.

Policy

These standards must be used in conjunction with all other referenced standards, and when considered in isolation from the referenced standards may not constitute adequate conformance.

Conflict of Information or Clarification

Whenever a conflict of information occurs or clarification of instruction is required, all queries shall be made to Deakin University eSolutions (DeS).

1 Unit Testing

1.1 Web applications should employ Unit Testing where practical

1.2 When to write unit tests

A unit test can be written:

- **Before** any code is written. I.e.: “Test Driven Development” (http://en.wikipedia.org/wiki/Test_driven_development) OR
- **During** the build phase OR
- **After** the build phase

However, a combination of all three may be beneficial.

1.3 Test coverage

The unit test defines a “contract” of the functionality of a class and must be comprehensive enough so as to prove that the methods being tested are working as intended, and that the tests exercise the method’s preconditions.

1.4 Tests must be written in a way that specific and strict.

A unit test must be written in a way that only valid and expected values allow the unit test to pass.

1.5 Tests must be automated

Any unit tests written must be completely automated. That is, no human interaction should be required between the time the test is run and the time the test results are produced.

1.6 Tests must be isolated

Tests must be fully independent of each other. The outcome of one test must not have an effect on the outcome of other tests.

1.7 Tests should take a micro view of the architecture

There should be multiple tests with each testing one small, atomic functionality. A test should not test more than one function.

1.8 Unit Tests shall be created in suites

Unit Tests shall be created in suites (<https://phpunit.de/manual/current/en/phpunit-book.html#organizing-tests.filesystem>) and organised in a hierarchy. There should be one test suite per “package”.

1.9 Test class and method naming shall conform with PHPUnit conventions

<https://phpunit.de/manual/current/en/phpunit-book.html#organizing-tests.filesystem>

1.10 Tests must be kept up to date

If any additional functionality is added to a class, a new set of test cases must be written to test that functionality.

1.11 Unit tests must be executed prior to application deployment

Additionally, it is also advisable to run tests at various stages throughout the modification of a class.

2 Security testing

2.1 The OWASP Testing guide should be used

http://www.owasp.org/images/5/56/OWASP_Testing_Guide_v3.pdf