
CONTENTS

1 ABOUT AUTHOR:	6
2 INTRODUCTION TO TEST DRIVEN DEVELOPMENT	7
3 PART 1	8
3.1 Cycle: Red-Green-Refactor	8
3.2 Black-box testing and White-box testing:	9
3.2.1 White-box Testing:	9
3.2.2 Black-box Testing:	9
3.2.3 The grey box :	10
3.3 JUnit:	10
3.4 JUnit Test Class:	11
3.5 Test driven development using with Junit:	12
3.6 Refactoring Unit Tests:	23
3.7 Code Coverage:	23
4 PART 2	27
4.1 Junit:	27
4.1.1 Structure	27
4.2 Example	30
4.2.1 Exception testing	36
4.3 MUTATION TEST WITH PIT	41
4.4 Mocking	44
4.4.1 Mockito	44
4.5 Mockito - Adding Behavior	53
4.6 Mockito - Verifying Behavior	57
4.7 Mockito - Expecting Calls	62
4.8 Mockito - Varying Calls	66
4.9 Mockito - Exception Handling	70
4.10 Mockito - Create Mock	74
4.11 Mockito - Ordered Verification	76
4.12 Mockito - Callbacks	80
5 PART 3	86
5.1 Maven:	86
5.1.1 Maven: Installation	88
5.1.2 Import a Maven project in Eclipse	89
5.1.3 Create a Maven project from Eclipse using an archetype	89
5.1.4 Java settings	91

7.2.1 The Dockerfile	153
8 DOCKER CONTINUE	159
8.0.1 Build the Docker image from Maven	159
9 INTEGRATION TESTS	164
9.0.1 Example	165
9.1 Unit tests with databases	169
9.1.1 Integration tests	172
9.1.2 Source folder for integration tests	172
9.1.3 Docker and Testcontainers	173
9.1.4 Tests with Maven	175
9.1.5 tests with Docker and Maven	177
9.1.6 integration tests in Travis CI	182
10 GRAPHICAL USER INTERFACE TESTING	183
10.1 What do you Check-in GUI Testing?	183
10.1.1 Example GUI Testing Test Cases	184
10.1.2 Challenges in GUI Testing	184
10.1.3 GUI Testing Tools	185
10.2 Consider using a BDD framework	185
11 END-TO-END TESTING	195
11.1 Why End to End Testing	195
11.1.1 End to End Testing Process:	195
11.1.2 How to create End-to-End Test Cases?	196
12 PART 4	198
12.1 Code Quality-SonarQube :-	198
12.1.1 Key Aspects to Measure	198
12.1.2 How to Improve Code Quality	199
12.2 SonarQube	201
12.3 Key concepts in Static Code Analysis	202
12.3.1 Write Clean Code	203
13 JAVA PROJECT USING TEST DRIVEN DEVELOPMENT	207
14 CONCLUSION	208

1

ABOUT AUTHOR:

Joseph Thachil George is a Technical Consultant for International Game Technology (IGT), Rome, Italy. Additionally, Joseph is pursuing doctorate (PhD) in Computer Science and Engineering at the University of Lisbon, Portugal. He has completed M. S in Cyber Security from the Università degli Studi di Firenze, Italy. In addition, he is also part of the research group (DISIA) of the University of Florence, Italy, and the research group (INESC-ID Lisbon) of the University of Lisbon, Portugal.

His research interests cover Dynamic Malware Analysis, Blockchain technology- Hyperledger fabric, and cyber security. He published five books *Cybercrime and Social Media Relationships*, *Designing Distributed Systems* , *Social Network Analysis* ,*Advanced Distributed Systems Design* and *Network Security Management* respectively. In IGT he is been a part of various project related to game configuration and integration in various platform. Specialized in Java and spring boot-based projects.

He has also worked in various companies in India, Angola, Portugal, and UK. In total he has seven years of experience in various IT companies.