قال تعالى:

وَ قُلِ اعْمَلُواْ فَسَيَرَى اللّهُ عَمَلَكُمْ

وَ رَسُولُهُ وَ الْمُؤْمِنُونَ وَ سَتُرَدُّونَ إِلَى عَالِمِ الْغَيْبِ وَ الشَّهَادَةِ

فَيُنَبِّئُكُم بِمَا كُنتُمْ تَعْمَلُونَ)

سورة التوبة الآية (105).

ABSTRACT

Testing is one of the most important parts of the software life cycle. Testing reduces the cost of development by minimizing revision and increases customer satisfaction by insuring that the system is of high quality before delivery. The art of testing is revolves around designing effective test cases. A test case is a set of conditions generated from system requirements to verify the expected outputs/results against actual outputs/results. Test case design and generation has always been part of the testing process, but their fundamental significance was only recognized once testing was automated, such as in keyword-based testing. Automating the whole testing process has great benefits such as increasing capacity, reducing test execution costs, and reducing overall testing time. MDA (Model Driven Architecture) is an OMG(Object Management Group) standard development method. Models of different systems are structured explicitly into the Platform Independent Model (PIM) and the Platform Specific Model (PSM).This thesis proposes a new test execution methodology in MDA Application by using UML(Unified Modeling Language) And MOF(Meta Object Facility) operations (Create, Read, Update, delete) within an RDBMS(Relational Database Management System) model instance. The results of this work is test executing approach for test cases that generated from PSM model instance using MOF operation (Create, Read, Update, delete).This results make reading, parsing test cases files more easy because it's write in XML(Extensible Markup Language) format .

المستخلص

الإختبار هو واحد من اهم اجزاء دورة حياة البرمجيات. يقلل الاختبار من تكاليف التطوير عن طريق توفير الجهد في اعادة العمل مرة اخرى، يرفع من رضا العملاء عن النظام عن طريق التأكد من ان جودة النظام عالية قبل عملية التسليم للعملاء. فن الاختبار يدور حول تصميم حالات اختبار فعالة. وحالة الاختبار هي مجموعة من الشروط تتولد من متطلبات النظام للتحقق من المخرجات المتوقعة او النتائج المتوقعة بالمقارنة مع المخرجات الفعلية او النتائج الفعلية.

تصميم و توليد حالات الاختبار هما دائما جزءا من عملية الاختبار. ولكن اهميتهما الاساسية ظهرت عندما اصبح الاختبار الياً. مثال في طريقة الاختبار القائمة على الكلمة. أتمتة جميع عمليات الاختبار لها فوائد عظيمة مثل إمكانية تنفيذ المزيد من الاختبارات. تقليل تكلفة تنفيذ الاختبار وتقليل الزمن الكلي للاختبار.

المعمارية القائمة على النموذج هي مقياس طريقة تطوير من مجموعة إدارة الكائنات. النماذج المختلفة للانظمة تتوضح في شكل نموذج مستقل ونموذج معين.

هذة الاطروحة تقترح منهجية جديدة لتنفيذ اختبار في تطبيقات المعمارية القائمة على النموذج عن طريق استخدام لغة النمذجة الموحدة و عمليات لغة عالية المستوى لنمذجة العناصر (إنشاء ،قراءة،تعديل،إلغاء) منطبقة على مثال من نموذج نظام قاعد بيانات علائقية.

نتائج هذا العمل هو طريقة تنفيذ حالات الاختبار المتولدة من مثال نموذج معين بإستخدام عمليات لغة عالية المستوى لنمذجة العناصر (إنشاء ،قراءة،تعديل،إلغاء). هذه النتائج تجعل عملية قراءة وترجمة ملفات حالات الاختبار أكثر سهولة لانها مكتوبة في شكل لغة التوصيف الموسعة.

Table of Contents

Abstract ..... **ii**

Table of content iv

List of figures vi

List of abbreviation vii

Chapter 1 ..... **1**

Introduction 1

Problem 2

Objectives 2

Importance of research 2

Scope 3

Chapter 2 ..... **4**

Introduction 4

Testing in software Engineering 4

Test cases 5

How to Generate Test cases 5

Coverage criteria 6

Association-End Multiplicity(AEM) Coverage criteria 6

Model Based Testing(MBT) 8

Model Driven Architecture (MDA) 10

Platform Independent Model (PIM) 10

Platform Specific Model (PSM) 11

Mapping PIM to PSM 11

Test Execution 13

Approaches to mapping tests 14

The Adaptation Approach 14

The Transformation Approach 15

The mixed approach 15

The comparisons among Execution approaches 15

Related Work 16

Chapter 3 ..... **18**

Introduction 18

Methodology 18

How to get Oracles 19

Case Study: system of mapping UML to RDBMS 21

Case study RDBMS system Instance (Student Information System) 24

Connect PSM instance to Platform using (CRUD) 24

Execute Test cases using (CRUD). 25

How to get Oracles 27

Discussion 29

Chapter 4 ..... **29**

Conclusions 30

References 31

LIST OF FIGURES

2.1	Student Information system	7
2.2	Model Based Testing steps	9
2.3	PIM Example	12
2.4	PSM Example	12
2.5	MDA Meta-model	13
2.6	Test Execution Approaches	14
3.1	Test case file Example	19
3.2	Output Element in Test case file	20
3.3	UML Meta-Model (PIM)	22
3.4	RDBMS Meta-Model (PSM)	23
3.5	Simple RDBMS Meta-Model	24
3.6	Test case Example	26
3.7	PSMinstance.xmifile	28

LIST OF ABBREVIATIONS

MDA	Model Driven Architecture
OMG	Object Management Group
MBT	Model-Based Testing
MOF	Meta Object Facility
PIM	Platform Independent Model
PSM	Platform Specific Model
QVT	Query/Views/Transformations
SUT	System Under Test
API	Application Programming Interface
EMF	Eclipse Modeling Framework
IEEE	Institute of Electrical and Electronics Engineers
UML	Unified Modeling Language
UML2	Unified Modeling Language version 2
XML	Extensible Markup Language
XMI	XML Metadata Interchange
V&V	Verification and Validation
CRUD	Create, Read, Update, Delete
RDBMS	Relational Database Management System
Reg	Registration