

Índice

CAPITULO 1. INTRODUCCIÓN	3
1.1 Introducción y Definición del Problema.....	3
1.2 Objetivo general	4
1.3 Objetivos específicos.....	4
1.4 Alcances y limitaciones	4
1.5 Hardware y Software a utilizar.....	5
1.6 Organización del Documento.....	5
CAPITULO 2. MARCO TEÓRICO	7
2.1 Patrones de Diseño.....	7
2.2 “Test Driven Development”	12
CAPITULO 3. “TEST DRIVEN DEVELOPMENT”	15
3.1 Uso de JUnit como “framework” para realizar pruebas unitarias	15
3.2 Noción Básica del Uso de “Test Driven Development” como “framework”	17
3.3 Características de JUnit.....	26
3.4 Limitaciones del “Framework”.....	27
CAPITULO 4. PATRONES DE DISEÑO	28
4.1 Patrón “Bridge”	28
4.2 Patrón “Abstract Factory”	29
4.3 Patrón “Builder”	30
4.4 Patrón “Iterator”	30
4.5 Patrón “Command”.....	31
4.6 Patrón “Strategy”	32
4.7 Patrón “Decorator”	33
4.8 Patrón “Flyweight”.....	34

4.9 Patrón “Chain of Responsibility”	35
4.10 Patrón “Proxy”	36
4.11 Patrón “Interpreter”	37
CAPITULO 5. DISEÑO E IMPLEMENTACIÓN DEL “SQL INTERPRETER”	39
5.1 Arquitectura del Sistema	39
5.2 Capa de Almacenamiento de Datos.....	41
5.2.1 Patrón “Bridge Pattern”	41
5.2.2 Patrón “Abstract Factory”	43
5.2.3 Patrón “Builder”	47
5.2.4 Patrón “Iterator”	50
5.2.5 Patrón “Command”	54
5.2.6 Patrón “Strategy”	55
5.2.7 Patrón Decorador	58
5.3 Capa del Motor “SQL”	59
5.3.1 Patrón “Flyweight”	59
5.3.2 Patrón “Chain of Responsibility”	63
5.3.3 Patrón “Proxy”	65
5.3.4 Patrón “Interpreter”	67
CAPITULO 6. CONCLUSIÓN	77
6.1 Trabajo a Futuro	79
APÉNDICE A. PROTOTIPO SQL INTERPRETER	81
APÉNDICE B. TUTORIAL DEL USO DE JUNIT 3.8.1 Y NETBEANS 5.5	83
Creación una clase de prueba JUnit con Netbeans 5.5.....	83
Ejecución de las pruebas unitarias mediante JUnit	87
REFERENCIAS	90