

Оглавление

Введение	6
1. Жизненный цикл программного обеспечения	7
1.1. Основные понятия	7
1.2. Процесс анализа требований к программным средствам	9
1.3. Процесс проектирования архитектуры программных средств	10
1.4. Процесс детального проектирования программных средств	11
1.5. Процесс конструирования программных средств	12
1.6. Процесс комплексирования программных средств	13
1.7. Процесс квалификационного тестирования программных средств	14
1.8. Поставка и внедрение	15
1.9. Сопровождение программного продукта	16
1.10. Модели жизненного цикла	17
2. Методологии разработки программного обеспечения	21
2.1. Классические методологии	21
2.1.1. Метод функциональной декомпозиции	21
2.1.2. Метод анализа потоков данных	23
2.2. Объектно-ориентированный подход	24
2.3. <i>Agile</i> -методики	24
3. Языки UML и OCL	27
3.1. Общие понятия	27
3.2. Диаграмма вариантов использования	28
3.3. Диаграмма классов	33
3.4. Диаграмма состояний	37
3.5. Диаграмма деятельности	40
3.6. Диаграмма последовательностей	42
3.7. Диаграмма кооперации	46
3.8. Диаграмма компонентов	46
3.9. Диаграмма развертывания	47
3.10. Язык OCL	48
4. Выполнение этапов анализа и проектирования на языке UML	50
4.1. Основные положения	50

4.2. Определение требований.....	53
4.3. Уточнение и структурирование требований.....	57
4.4. Проектирование архитектуры.....	60
4.5. Проектирование классов и подсистем.....	63
4.6. Проектирование вариантов использования.....	65
4.7. Использование среды <i>Microsoft Visual Studio</i> для выполнения анализа ...	67
4.8. Создание диаграмм проектирования.....	69
5. Реализация на языке C# в среде <i>Microsoft Visual Studio</i>	76
5.1. Простейший пример.....	76
5.1.1. Постановка задачи.....	76
5.1.2. Средства управления работой программы.....	77
5.1.3. Создание меню.....	78
5.1.4. Ввод/вывод массивов.....	79
5.1.5. Форматированный ввод/вывод двумерного массива.....	83
5.2. Создание многооконных приложений.....	87
5.2.1. Создание SDI-приложения.....	88
5.2.2. Создание MDI-приложения.....	91
5.3. Реализация алгоритмов.....	96
5.4. Работа с библиотекой классов.....	98
5.5. Реализация взаимодействия с базой данных.....	101
5.5.1. Структура базы данных.....	101
5.5.2. Структура проекта.....	104
5.5.3. Просмотр данных без изменений.....	105
5.5.4. Просмотр и изменение данных.....	107
5.5.5. Извлечение данных.....	108
5.5.6. Изменение данных хранимыми процедурами.....	110
5.5.7. Использование данных из базы для вычислений.....	113
5.6. Использование стандартных классов в реализации.....	115
5.6.1. Работа с классом <i>List</i>	116
5.6.2. Работа со словарем — классом <i>Dictionary</i>	123
5.6.3. Список из списков.....	127
5.6.4. Использование диалоговых окон для работы со стандартными классами.....	129
5.7. Дополнительные средства создания интерфейса пользователя.....	136

5.8. Вывод на печать	142
5.9. Рефакторинг	145
5.10. Работа в <i>WPF</i>	150
5.10.1. Простейший пример	151
5.10.2. Работа с двумерным форматизированным массивом	154
5.10.3. Работа со списком	156
6. Тестирование программного обеспечения	160
6.1. Методы проверки программного обеспечения	160
6.2. Тестирование примитивных программ	162
6.2.1. Тестирование «черного ящика»	162
6.2.2. Тестирование «белого ящика»	168
6.2.3. Метод функциональных диаграмм	171
6.2.4. Предположение об ошибке	174
6.3. Тестирование программных комплексов, построенных методом функциональной декомпозиции	174
6.4. Тестирование программных комплексов, построенных по объектно-ориентированной методике	177
6.5. Отладка программ	180
6.6. Средства тестирования <i>Microsoft Visual Studio</i>	181
6.7. Современный подход к проверке	190
Заключение	193
Библиографический список	194