

ИСПОЛЬЗОВАНИЕ АРХИТЕКТУРЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ПОДГОТОВКЕ КАДРОВ ИНФОРМАТИЗАЦИИ

Л.В. Яковлева

Россия, г. Мурманск

В настоящее время Web-программирование играет важную роль в сфере разработки программного обеспечения (ПО). Повсеместное использование Web-приложений является важной тенденцией развития компьютерных технологий, операционных систем, сетевых архитектур и прикладных программ. Развитие информационных технологий на современном этапе требует дальнейшего совершенствования подготовки специалистов в области создания программного обеспечения, разработка которого характеризуется сложностью программных систем. Поэтому очень важно изучение дисциплины “Архитектура ПО“ студентами и специалистами в области разработки ПО, так как это позволит специалистам в области проектирования ПО получить профессиональные навыки в эффективной разработке ПО. В недавнем прошлом проблемы сложности решались разработчиками путем правильного выбора структур данных, разработки алгоритмов, технологий быстрой разработки приложений (Rapid Application Development) . В настоящий момент, когда в области архитектуры и дизайна приложений постоянно происходят изменения, актуальным является правильный выбор архитектурного типа приложения для уменьшения времени разработки, стоимости и сложности разработки. Применение архитектурных шаблонов способствует реализации гибкого дизайна приложения и сквозной функциональности, характеризующихся слабым связыванием, что делает возможным повторное использование и упрощает поддержку.

Особенно хотелось бы отметить, применение в данной дисциплине методов аспектно-ориентированного программирования (АОП). АОП – один из новых подходов к программированию, предназначенный для модуляризации

сквозной функциональности и ее автоматизированного, безопасного и надежного добавления в целевую программу, а также модификации в целевой программе некоторой уже реализованной сквозной функциональности. Одной из реализаций АОП для платформы Microsoft .NET является система Aspect.NET.

В данной дисциплине подробно рассматриваются вопросы опережающего тестирования и принципы гибкого проектирования: принцип единственной обязанности, принцип открытости/закрытости, принцип подстановки Лисков, принцип инверсии зависимости, принцип разделения интерфейсов.

В разработке важную роль играют показатели качества разработанного ПО. Для оценки качества разработанного ПО используются метрики: сцепленность по связям, входящая и исходящая связность, абстрактность, неустойчивость. Расчет метрик можно автоматизировать с помощью функционального языка (Ruby).

В итоге, после изучения курса “Архитектура ПО“ студенты могут правильно выбрать стратегии и шаблоны проектирования, определить ключевые технические решения, реализовать основные показатели качества Web-приложений.

Литература

1. Роберт Мартин, Мика Мартин. Принципы, паттерны и методики гибкой разработки на языке C#.
2. Aspect J Home Page. <http://www.aspectj.org>