



Автономная некоммерческая организация
дополнительного профессионального образования
(АНО ДПО «Инфосфера»)

Центр профессиональной подготовки
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ

**Рабочая программа дисциплины
«Объектно-ориентированное проектирование»**

Разработал:
преподаватель ИПС
АНО ДПО «Инфосфера»
А.Н. Малов

Йошкар-Ола, 2017

Пояснительная записка

Целью курса являются: изучение методов объектно-ориентированного анализа и проектирования; изучение приемов разработки программных приложений, ориентированных на повторной коде (методы повторного использования); знакомство с унифицированным языком моделирования (UML); изучение типовых приемов проектирования (паттернов проектирования).

Планируемые результаты обучения

Осуществляется предварительная самостоятельная или под руководством разработчика алгоритмов с использованием графических средств (блок-схемы, UML-диаграммы и др.). Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.

Выполняются самостоятельная разработка процедур сборки модулей и компонент программного обеспечения и верификация выпусков программного продукта. Производится разработка процедур развертывания и обновления программного обеспечения, процедур миграции и преобразования (конвертации) данных и программных интерфейсов с использованием выбранных программных средств, технологий создания открытых систем. Осуществляется решение различных типов задач проектирования программных комплексов различной сложности, выбор способов реализации взаимодействия программных компонент/модулей. Требуется взаимодействие с программистами-разработчиками модулей, архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

В процессе интеграции программных модулей и компонент и верификации выпусков программного продукта осуществляется сборка модулей и компонент программного обеспечения, производится интеграция с внешней средой. Обеспечивается согласованное функционирование и требуемый уровень качества.

Проведение интеграции программных модулей и компонент и верификация выпусков программного продукта предполагают определение задач программной интеграции, распределение задач между подчиненными, обеспечение взаимодействия подчиненных сотрудников.

Программист несет ответственность за результат выполнения работ на уровне группы программистов. В процессе интеграции требуется взаимодействие с архитектором

программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

Учебно-тематический план

№	Наименование разделов	Всего часов	В том числе		Форма контроля
			Лекции	Практ. занятия	
1	Введение в объектно-ориентированный анализ и проектирование	12	6	6	Защита лабораторной работы
2	Унифицированный язык моделирования UML	12	6	6	Защита лабораторной работы
3	Введение в паттерны проектирования Порождающие паттерны	12	6	6	Защита лабораторной работы
					Контрольная работа
4	Структурные паттерны	12	6	6	Защита лабораторной работы
					Контрольная работа
5	Введение в объектно-ориентированный анализ и проектирование	12	6	6	Защита лабораторной работы
6	Унифицированный язык моделирования UML	12	6	6	Защита лабораторной работы
					Контрольная работа
	Итого	72	36	36	

Содержание курса

Тема 1. Введение в объектно-ориентированный анализ и проектирование.

Предмет объектно-ориентированного анализа и проектирования. Понятие повторного использования проектных решений. Основные концепции ООП с точки зрения повторного использования. Влияние повторного использования на механизмы современных языков программирования. Инкапсуляция, наследование, полиморфизм. Достоинства и недостатки. Элементарные приемы объектного проектирования: композиция и делегирование. Разница между объектным и структурным подходами.

Тема 2. Унифицированный язык моделирования UML.

Что такое UML. Основные концепции Унифицированного языка моделирования: визуализация, прямое и обратное проектирование, специфицирование, документирование. Виды диаграмм UML: диаграмма классов, диаграмма активностей, диаграммы взаимодействия, диаграммы вариантов использования. Обзор CASE-средств для построения диаграмм UML.

Тема 3. Введение в паттерны проектирования.

Что такое типовое решение проектирования. Пример паттерна проектирования: Модель-вид-контроллер. Форматы описания паттернов проектирования. Методика применения типовых решений проектирования.

Тема 4. Порождающие паттерны.

Обзор применения порождающих типовых решений проектирования. Порождающие паттерны: абстрактная фабрика, строитель, фабричный метод, прототип, одиночка. Обсуждение порождающих паттернов проектирования.

Тема 5. Структурные паттерны.

Обзор применения структурных решений проектирования. Структурные паттерны: адаптер, мост, компоновщик, декоратор, фасад, заместитель, приспособленец. Обсуждение структурных паттернов проектирования.

Тема 6. Поведенческие паттерны.

Обзор применения поведенческих решений проектирования. Поведенческие паттерны: цепочка обязанностей, команда, интерпретатор, итератор, медиатор, хранитель, посредник, состояние, стратегия, фабричный метод, посетитель. Обсуждение поведенческих паттернов проектирования.

Пособия по изучению курса

1. Анализ требований и создание архитектуры решений на основе Microsoft .NET: Офиц. пособие Microsoft для самост. подготовки: Учеб курс MCSD. – М.: Русская редакция, 2004. – 383 с.
2. Арлоу Д., Нейштадт А. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование, 2-е издание.– Пер. с англ. – СПб: Символ-Плюс, 2007.– 624 с.
3. Блаха М., Рамбо Дж. UML 2.0: Объектно-ориентированное моделирование и разработка – 2-е изд. – СПб.: Питер, 2007. – 544 с.
4. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования – Питер, 2003.
5. Крачтен Ф. Введение в Rational Unified Process. – 2-е изд. – М.: Изд. дом «Вильямс», 2002. – 240 с.
6. Ларман К. Применение UML 2.0 и шаблонов проектирования (третье издание). – М.: Вильямс, 2007.
7. Мейер Б. Объектно-ориентированное конструирование программных систем: Пер. с англ. – М.: Рус. редакция, 2005. – 1198 с.
8. Поллис Г., Огастин Л., Лоу К., Мадхар Д. Разработка программных проектов на основе Rational Unified Process (RUP). – М.: БИНОМ, 2005. – 255 с.
9. Ройс У. Управление проектами по созданию программного обеспечения: Унифицированный подход. – М.: ЛОРИ, 2002. – 424 с.
10. Фанг Д., Йу К., Лай К. Введение в IBM Rational Application Developer : Учеб. руководство: Пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2006. – 564 с.
11. Фаулер М. UML. Основы, 3-е издание. – Пер. с англ. – СПб: Символ-Плюс, 2004. – 192 с.
12. Фаулер М. Архитектура корпоративных программных приложений. – М.: Изд. дом «Вильямс», 2004. – 544 с.
13. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. –СПб.: Питер, 2002.

Контрольные задания

Темы индивидуальных заданий имеют общий шаблон: Спроектировать применение и оценить влияние на архитектуру паттерна <pattern> в системе <project>, где <pattern> –

название конкретного паттерна объектно-ориентированного проектирования или архитектурного решения, <project> – некоторая документация по реальному проекту, включающая всю информацию, необходимую для решения задачи.

Темы группового проектирования формируются по следующему сценарию.

Дано: краткое описание предметной области, контактная информация заказчика/пользователя/эксперта (одного из них или нескольких).

Требуется: спроектировать архитектуру приложения.

Дополнительно: детализировать и реализовать проект.

Вопросы и задания для текущего контроля:

Язык UML. Общая характеристика.

Основные элементы языка UML.

Общая характеристика порождающих типовых приемов проектирования.

Общая характеристика структурных приемов проектирования.

Общая характеристика поведенческих типовых приемов проектирования.