

Аннотация рабочей программы дисциплины

Б1.О.23 «Объектно-ориентированное программирование и шаблоны проектирования»

Курс 3 Семестр 5 Количество з.е. 2

Объем трудоемкости: 2 зачетных единиц (72 ч., из них – 50 час. аудиторной нагрузки: лекционных 16 ч., лабораторных работ - 34 ч., 19,8 часов самостоятельной работы, 2 часов КСР, 0,2 часа ИКР.), форма контроля – зачет.

Цель дисциплины: изучение объектно-ориентированной парадигмы программирования и отработка навыков ОО проектирования.

Задачи дисциплины:

1. Изучение объектно-ориентированной парадигмы программирования.
2. Изучение языка программирования Java
3. Получение практического опыта ОО проектирования приложения.
4. Получение практического опыта ОО проектирования предметной области
5. Изучение ООП паттернов проектирования и архитектурных паттернов.
6. Научиться строить описание архитектуры приложения с помощью диа-граммы классов и диаграммы последовательностей
7. Отработка умений реализации заданного архитектурного паттерна.
8. Отработка навыков построения web-сервиса средствами JVM языков на примере фреймворков Spring, SpringBoot.

Место дисциплины в структуре ООП ВО:

Дисциплина «Объектно-ориентированное программирование и шаблоны проектирования» относится к базовой части блока Б1 Дисциплины (модули).

Дисциплина в значительной степени **взаимодействует для формирования компетенций** с дисциплинами:

- Бэкенд-разработка;
- Кроссплатформные десктоп приложения;

Дисциплина является базовой для дисциплин: микросервисная архитектура, системы искусственного интеллекта, разработка мобильных приложений.

- Требованиям к «входным» знаниям является знания основных алгоритмов и структур данных, понимание основ веб-разработки, разработки десктопных приложений.

Результаты обучения (знания, умения, опыт, компетенции):

Содержание и структура дисциплины:

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций: ОПК-2.1; ОПК-2.2; ОПК-5.1, РЛ-2.1

ОПК-2	<i>Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</i>
ОПК-2.1	Способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к реализации алгоритмов решения прикладных задач

	<p>Знает стандарт UML2.5 для описания архитектуры веб-сервиса и ООП архитектуры предметной области</p> <p>Знает структуру технического задания</p> <p>Умеет реализовать сервис по требованиям в команде</p> <p>Умеет применять методы рефакторинга и обнаруживать антипаттерны в коде</p>
ОПК-2.2	<p>Применяет современный математический аппарат при построении моделей в различных областях человеческой деятельности</p> <p>Владеет способностью составить модель предметной области</p>
ОПК-5	<i>Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</i>
ОПК-5.1	<p>Аргументировано применяет методы проектирования, разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности</p> <p>Знает паттерны проектирования и архитектурные паттерны</p> <p>Умеет применять методы рефакторинга и обнаруживать антипаттерны в коде</p> <p>Владеет способностью разрабатывать веб сервисы на Java Spring, SpringBoot</p> <p>Владеет способностью реализовать MVC приложения с нуля без фреймворков</p>
PL-2	<i>(П) Способен применять JVM-совместимые языки программирования для решения задач в области ИИ</i>
PL-2.1	<p>Разрабатывает и отлаживает прикладные решения разного уровня сложности и для широкого круга конечных пользователей с использованием JVM-совместимых языков программирования, тестирует, испытывает и оценивает качество таких решений</p> <p>Применяет основные библиотеки для решения рутинных задач в серверном программировании: ввод-вывод, применение простейших примитивов многопоточного программирования, интеграция с базами данных.</p> <p>Понимает модель памяти Java и способен поддерживать приложения с высоким параллелизмом и конкуренцией. Понимает алгоритмы сборки мусора и способен оптимизировать сборку мусора.</p> <p>Имеет представление о том, как построить взаимодействие со специализированными хранилищами данных эффективно, через инструменты разработчика для JVM-совместимых языков</p>

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
 Разделы дисциплины, изучаемые в _5_ семестре (очная форма)

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Введение в разработку Java	16	4		8	4
2.	MVC приложение	16	4		8	4
3.	Рефакторинг и паттерны проектирования	16	4		8	4
4.	Архитектурные паттерны	18	4		10	4
ИТОГО по разделам дисциплины		66	16		34	16
Контроль самостоятельной работы (КСР)		2				

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
	Промежуточная аттестация (ИКР)	0,2				
	Подготовка к текущему контролю	3,8				
	Общая трудоемкость по дисциплине	72				

Примечание: Л – лекции, КСР – контрольные и самостоятельные работы, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

Курсовые проекты или работы.

Не предусмотрены учебным планом

Вид аттестации: ЛР, контрольная работа, зачет.

Автор А.С.Жук, доцент КВТ