

## Аннотация рабочей программы дисциплины

### Б1.В.06 «ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ»

Направление подготовки/специальность 02.03.02 Фундаментальная информатика и информационные технологии

Курс 3 Семестр 5 Количество з.е. 4

**Объем трудоемкости:** 4 зачетных единицы (144 часа, из них – 82,3 часа аудиторной нагрузки: лекционных 34 ч., лабораторных работ - 34 ч., 4 часа КСР, 0,3 часа ИКР, 35,7 часов подготовки к экзамену, 10 часов курсового проектирования, 26 часов самостоятельной работы).

**Цель дисциплины:** Целью преподавания и изучения дисциплины «Паттерны проектирования» является изучение теоретических и практических основ разработки ООП приложений в рамках заданной архитектуры.

#### **Задачи дисциплины:**

Результатом освоения дисциплины «ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ» является приобретение следующих знаний и умений:

студент должен **знать** теоретические принципы программирования в ООП парадигме, классификацию и базовые принципы применения паттернов проектирования, основные признаки плохого кода и методы рефакторинга, принципы реализации базовых архитектур ООП приложений в рамках заданных фреймворков, принципы SOLID ООП разработки; **уметь** создавать приложения в рамках архитектурного паттерна MVC и производных от него паттернов без использования фреймворков, проводить рефакторинг кода, составлять ООП архитектуру приложения, разбирать аспекты построения заданного архитектурного паттерна в рамках конкретного фреймворка на примере MVC паттерна в фреймворке ruby-on-rails, применять принципы SOLID ООП разработки там, где это необходимо; **владеть** методами рефакторинга, принципами построения ООП архитектуры без антипаттернов; паттернами проектирования и аспектами построения современных архитектурных паттернов.

#### **Место дисциплины в структуре ООП ВО:**

Дисциплина «ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ» относится к вариативной части обязательных дисциплин учебного курса бакалавриата.

Для изучения дисциплины необходимо знание основ программирования, технологий баз данных, конструирования алгоритмов и структур данных, разработки в компьютерных сетях. Знания, получаемые при изучении дисциплины «ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ», используются при изучении других дисциплин учебного плана бакалавра (кроссплатформенное программирование, программирование для мобильных платформ, бек-разработка, распределенные задачи и алгоритмы, принципы командной разработки ПО), а также при работе над курсовыми работами и выпускной работой.

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

| № п.п. | Индекс компетенции | Содержание компетенции (или ее части)   | В результате изучения учебной дисциплины обучающиеся должны |                                        |                                   |
|--------|--------------------|-----------------------------------------|-------------------------------------------------------------|----------------------------------------|-----------------------------------|
|        |                    |                                         | знать                                                       | уметь                                  | владеть                           |
| 1.     | УК-2               | Способен определять круг задач в рамках | основные признаки плохого кода и методы                     | проводить рефакторинг кода, составлять | методами рефакторинга, принципами |

|    |      |                                                                                                                                                                                                                                                                 |                                                                                                                                                                                                                                                    |                                                                                                                                                                                                                                                                                                                                             |                                                                                      |
|----|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
|    |      | поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений                                                                                                                              | рефакторинга                                                                                                                                                                                                                                       | ООП архитектуру приложения                                                                                                                                                                                                                                                                                                                  | построения ООП архитектуры без антипаттернов                                         |
| 2. | ПК-1 | Способен понимать и применять в научно-исследовательской и прикладной деятельности современный математический аппарат, основные законы естествознания, современные языки программирования и программное обеспечение; операционные системы и сетевые технологии; | теоретические принципы программирования в ООП парадигме, классификацию и базовые принципы применения паттернов проектирования, принципы реализации базовых архитектур ООП приложений в рамках заданных фреймворков, принципы SOLID ООП разработки. | создавать приложения в рамках архитектурного паттерна MVC и производных от него паттернов без использования фреймворков, разбирать аспекты построения заданного архитектурного паттерна в рамках конкретного фреймворка на примере MVC паттерна в фреймворке ruby-on-rails, применять принципы SOLID ООП разработки там, где это необходимо | паттернами проектирования и аспектами построения современных архитектурных паттернов |

### Содержание и структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.  
Разделы дисциплины, изучаемые в \_5\_ семестре (очная форма)

| № | Наименование разделов                   | Количество часов |                   |     |    |                      |
|---|-----------------------------------------|------------------|-------------------|-----|----|----------------------|
|   |                                         | Всего            | Аудиторная работа |     |    | Внеаудиторная работа |
|   |                                         |                  | Л                 | КСР | ЛР | СРС                  |
| 1 | 2                                       | 3                | 4                 | 5   | 6  | 7                    |
| 1 | Введение в ООП разработку на языке Ruby | 20               | 6                 |     | 10 | 4                    |

|   |                                         |      |    |    |    |    |
|---|-----------------------------------------|------|----|----|----|----|
| 2 | Базовые паттерны проектирования.        | 18   | 4  | 2  | 6  | 6  |
| 3 | Построение ООП приложений.              | 26   | 4  | 12 | 6  | 4  |
| 4 | Рефакторинг.                            | 12   | 4  |    | 4  | 4  |
| 5 | Архитектурные паттерны и принципы SOLID | 14   | 4  |    | 6  | 4  |
| 6 | Ruby-on-Rails.                          | 18   | 12 |    | 2  | 4  |
|   | Итого по разделам дисциплины            | 108  | 34 | 14 | 34 | 26 |
|   | Подготовка к экзамену                   | 35,7 |    |    |    |    |
|   | ИКР                                     | 0,3  |    |    |    |    |
|   | <i>Итого по дисциплине:</i>             | 144  |    |    |    |    |

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

### Курсовые проекты или работы.

Согласно учебному плану по данной дисциплине ведутся курсовые работы. Примерная тематика работ:

1. Компьютерные сети
  - 1.1. Автономный компьютеринг
  - 1.2. Одноранговые ad hoc сети
  - 1.3. Программные приложения для сети Интернет
  - 1.4. Поиск информации в сетях и ее анализ
2. Распределенные алгоритмы
  - 2.1. Системные распределенные алгоритмы
  - 2.2. Прикладные распределенные алгоритмы
  - 2.3. Мультиагентные системы
  - 2.4. Распределенные хранилища данных
  - 2.5. Распределенный сбор и анализ данных
3. Безопасные и надежные вычисления
  - 3.1. Криптографические преобразования и протоколы
  - 3.2. Надежность вычислений в стационарных и мобильных сетях
  - 3.3. Политики безопасности и прикладные логики
4. Алгоритмы интеллектуальной обработки данных
  - 4.1. Алгоритмы лингвистической обработки данных
  - 4.2. Алгоритмы компьютерного зрения и распознавания образов.
  - 4.3. Нейросетевые алгоритмы
5. Биоинспирированные алгоритмы
  - 5.1. Реализация процессов оптимизации вычислений.

**Вид аттестации:** зачет.

### Основная литература

1. Мацяшек, Л. А. Практическая программная инженерия на основе учебного примера=Practical software engineering. A case study approach : практическое пособие : [16+] / Л. А. Мацяшек, Брюс Ли Лионг ; пер. с англ. А. М. Епанешникова, В. А. Епанешникова. – 4-е изд., электрон. – Москва : Лаборатория знаний, 2020. – 959 с. – (Программисту). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=466728> (дата обращения: 29.05.2024). – Библиогр. в кн. – ISBN 978-5-00101-783-7. – Текст : электронный.
2. Фултон, Х. Программирование на языке Ruby : практическое пособие : [16+] / Х. Фултон ; пер. с англ. А. А. Слинкина. – 2-е изд. – Москва : ДМК Пресс, 2023. – 689 с. : ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=705236> (дата обращения: 29.05.2024). – ISBN 978-5-89818-531-2. – Текст : электронный.

3. Фултон, Х. Путь Ruby : практическое пособие : [16+] / Х. Фултон ; совм. с А. Арко ; пер. с англ. А. А. Слинкина. – 4-е изд. – Москва : ДМК Пресс, 2023. – 665 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=704224> (дата обращения: 29.05.2024). – ISBN 978-5-89818-334-9. – Текст : электронный.

Автор      Жук А.С. – старший преподаватель кафедры вычислительных технологий \_