

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
(ВлГУ)

УТВЕРЖДАЮ

Проректор  
по образовательной деятельности

  
А.А. Панфилов  
« 02 » \_\_\_\_\_ 09 2019 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**РЕФАКТОРИНГ ПРОГРАММНОГО КОДА**

**Направление подготовки:** 02.03.02 Фундаментальная информатика и информационные технологии

**Профиль/программа подготовки:** Мобильные и Интернет-технологии

**Уровень высшего образования:** бакалавриат

**Форма обучения:** очная (ускоренное обучение)

Семестр	Трудоемкость зач. ед./ час.	Лекции, час.	Практич. занятия, час.	Лаборат. работы, час.	СРС, час.	Форма промежуточной аттестации (экзамен/зачет/зачет с оценкой)
3	3 / 108	18	18	-	45	экзамен (27 ч.)
Итого	3 / 108	18	18	-	45	экзамен (27 ч.)

Владимир 2019

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины: познакомиться с теоретическими основами и методиками рефакторинга программного кода.

Задачи:

- приобретение теоретических знаний в области рефакторинга программного кода;
- приобретение практических навыков анализа и совершенствования программного кода, в первую очередь разработанного на основе объективно-ориентированного подхода;
- знакомство с инструментальными средствами, используемыми при рефакторинге программного кода.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Рефакторинг программного кода» относится к дисциплинам по выбору части, формируемой участниками образовательных отношений, блока Б1 Дисциплины (модули) учебного плана.

Пререквизиты дисциплины. Изучение данной дисциплины проходит в третьем семестре и опирается на результатах изучения дисциплин «Основы программирования», «Объектно-ориентированное программирование», «Архитектура компьютеров», «Иностранный язык».

## 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП

Код формируемых компетенций	Уровень освоения компетенции	Планируемые результаты обучения по дисциплине характеризующие этапы формирования компетенций (показатели освоения компетенции)
1	2	3
УК-1	частичное освоение	<p>Знать:</p> <ul style="list-style-type: none"><li>• базовые принципы системного анализа;</li><li>• основные философские понятия и теории, связанные с описанием устройства окружающего мира, а также их связь с законами и принципами развития, формулируемыми общественно-гуманитарными, естественными и техническими науками.</li></ul> <p>Уметь:</p> <ul style="list-style-type: none"><li>• выделять базовые составляющие задачи;</li><li>• осуществлять декомпозицию задачи;</li><li>• формулировать альтернативные подходы к решению задач в рамках выбранных видов профессиональной деятельности, в том числе на основе обобщения законов и методов различных наук, результатов из информационных источников.</li></ul> <p>Владеть:</p> <ul style="list-style-type: none"><li>• опытом использования индуктивного и дедуктивного подходов к решению задач;</li><li>• практическим опытом работы с информационными источниками.</li></ul>
ПК-1	частичное освоение	<p>Знать:</p> <ul style="list-style-type: none"><li>• методологии разработки программного обеспечения и технологии программирования;</li><li>• методы и средства проектирования программного обеспечения и баз данных;</li><li>• языки формализации функциональных спецификаций;</li><li>• принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения;</li><li>• типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при</li></ul>

		<p>разработке программного обеспечения;</p> <ul style="list-style-type: none"> <li>• основные концепции и атрибуты качества программного обеспечения;</li> <li>• стили написания кода;</li> <li>• методы классического системного анализа.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>• вырабатывать варианты реализации требований;</li> <li>• проводить оценку и обоснование рекомендуемых решений;</li> <li>• выбирать средства и варианты реализации программного обеспечения;</li> <li>• использовать существующие типовые решения и шаблоны проектирования программного обеспечения;</li> <li>• применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов;</li> <li>• формулировать цели, исходя из анализа проблем, потребностей и возможностей;</li> <li>• оценивать соответствие требованиям существующих систем и их аналогов.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>• навыками оценки возможностей, времени и трудоемкости реализации требований к программному обеспечению;</li> <li>• навыками проектирования структур данных, баз данных, программных интерфейсов;</li> <li>• навыками выбора стиля написания кода;</li> <li>• навыками анализа проблемной ситуации;</li> <li>• навыками согласования целей создания системы с заинтересованными лицами.</li> </ul>
ПК-3	частичное освоение	<p>Знать:</p> <ul style="list-style-type: none"> <li>• шаблоны проектирования слоёв или компонентов программного обеспечения;</li> <li>• стили написания кода;</li> <li>• принципы управления и мониторинга критически важных событий в мобильных и веб-приложениях;</li> <li>• методы и технологии анализа, проектирования и разработки мобильных и веб-приложений.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>• декомпозировать программные средства на компоненты;</li> <li>• выбирать программные компоненты;</li> <li>• применять шаблоны проектирования при реализации мобильных и веб-приложений;</li> <li>• определять программный интерфейс компонентов.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>• навыками оценки, выбора и проектирования однослойной и многослойной архитектуры мобильных и веб-приложений;</li> <li>• навыками выбора стиля написания кода.</li> </ul>

#### 4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 3 зачетных единицы, 108 часов

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Объем учебной работы, с применением интерактивных методов (в часах / %)	Формы текущего контроля успеваемости, форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	СРС		
1	Общие принципы и инструментальные средства рефакторинга программного кода.	3	1-5	4	6	–	12	6 / 60	рейтинг-контроль №1
2	Шаблоны проектирования и рефакторинга	3	6-11	8	8	–	17	8 / 50	рейтинг-контроль №2
3	Организация и оценка результатов рефакторинга	3	12-18	6	4	–	16	4 / 40	рейтинг-контроль №3
Всего за 3 семестр:		3	18	18	18	–	45	18 / 50	экзамен (27)
Наличие в дисциплине КП/КР		–	–	–	–	–	–	–	–
Итого по дисциплине		3	18	18	18	–	45	18 / 50	экзамен (27)

#### Содержание лекционных занятий по дисциплине

##### Раздел 1. Общие принципы и инструментальные средства рефакторинга программного кода.

- 1) Понятие рефакторинга. Цели рефакторинга и его место в жизненном цикле создания программного изделия. Инструментальные средства рефакторинга.
- 2) Методы рефакторинга.

##### Раздел 2. Шаблоны проектирования и рефакторинг.

- 3) История шаблонов (паттернов) проектирования. Общая методика создания и применения паттернов. Проблемы использования шаблонов проектирования. Классификация шаблонов проектирования. Основные шаблоны.
- 4) Структурные и порождающие шаблоны проектирования (Наблюдатель, Декоратор, Фабрика, Одиночка).
- 5) Шаблоны поведения (Состояние, Команда). Шаблон Модель-Представление-Контроллер (MVC).
- 6) Шаблоны рефакторинга (Коллекция идентичности объектов, Преобразователь данных, Загрузка по требованию, Единица работы).

##### Раздел 3. Организация и оценка результатов рефакторинга.

- 7) Определение необходимости в рефакторинге. Уровни рефакторинга. Организация проектной работы при рефакторинге программного кода.
- 8) Документирование процесса рефакторинга. Качественная и количественная оценка результатов рефакторинга.
- 9) Особенности рефакторинга мобильных и веб-приложений. Проблемы рефакторинга.

#### Содержание практических занятий по дисциплине

Во время практических занятий студенты осваивают общую методику рефакторинга, а также способы применения шаблонов проектирования, рассмотренных в рамках лекционного курса. В качестве основы занятий могут использоваться примеры программного кода, предоставленные преподавателем, либо результаты лабораторного практикума по дисциплине «Объектно-ориентированное программирование», изученной ранее.

## **Раздел 1. Общие принципы и инструментальные средства рефакторинга программного кода.**

- 1) *Анализ программного кода на необходимость рефакторинга. Выявление «запахов кода».*
- 2) *Выбор кода для рефакторинга.*
- 3) *Выбор методов рефакторинга.*

## **Раздел 2. Шаблоны проектирования и рефакторинг.**

- 1) *Выделение классов и интерфейсов.*
- 2) *Введение фабрики.*
- 3) *Применение шаблонов рефакторинга (часть 1).*
- 4) *Применение шаблонов рефакторинга (часть 2).*

## **Раздел 3. Организация и оценка результатов рефакторинга.**

- 1) *Составление плана рефакторинга.*
- 2) *Оформление плана рефакторинга.*

## **5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ**

В преподавании дисциплины «*Рефакторинг программного кода*» используются разнообразные образовательные технологии как традиционные, так и с применением активных и интерактивных методов обучения.

Активные и интерактивные методы обучения:

- Групповая дискуссия (все практические занятия);
- Разбор конкретных ситуаций (все лекционные и практические занятия);
- Уровневая дифференциация (контрольные мероприятия);
- Встречи с представителями фирм-разработчиков программного обеспечения (при изучении заключительного раздела дисциплины).

## **6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ**

**Текущий контроль успеваемости студентов**

### **Примерный перечень вопросов к рейтинг-контролю №1**

- 1) *Определение рефакторинга. Причины необходимости в рефакторинге.*
- 2) *Цели рефакторинга.*
- 3) *Роль рефакторинга в жизненном цикле разработки программного обеспечения.*
- 4) *Понятие «запахов кода». Признаки дурно пахнущего кода.*
- 5) *Причины появления «технического долга».*
- 6) *Раздувальщики кода.*
- 7) *Нарушители объектного дизайна.*
- 8) *Утяжелители изменений.*
- 9) *Замусориватели.*
- 10) *Опутыватели связями.*
- 11) *Инструментальные средства рефакторинга.*

### **Примерный перечень вопросов к рейтинг-контролю №2**

- 1) *Предпосылки появления методологии шаблонов проектирования.*
- 2) *Преимущества и недостатки методологии шаблонов проектирования.*
- 3) *Классификация шаблонов проектирования.*
- 4) *Шаблон Наблюдатель. Примеры использования.*
- 5) *Шаблон Декоратор. Примеры использования.*
- 6) *Шаблон Адаптер. Примеры использования.*
- 7) *Шаблон Фабрика. Примеры использования.*
- 8) *Шаблон Одиночка. Примеры использования.*
- 9) *Шаблон Абстрактная фабрика. Примеры использования.*
- 10) *Шаблон Команда. Примеры использования.*
- 11) *Шаблон Состояние. Примеры использования.*

- 12) Шаблон MVC.
- 13) Шаблон Коллекция идентичности объектов.
- 14) Шаблон Преобразователь данных.
- 15) Шаблон Загрузка по требованию.
- 16) Шаблон Единица работы.

### **Примерный перечень вопросов к рейтинг-контролю №3**

- 1) План мероприятий по организации рефакторинга.
- 2) Проверочный лист рефакторинга.
- 3) Составление методов.
- 4) Перемещение функций между объектами.
- 5) Организация данных.
- 6) Упрощение условных выражений.
- 7) Упрощение вызовов методов.
- 8) Решение задач обобщения.
- 9) Поддержание единства проекта при рефакторинге.
- 10) Документирование рефакторинга.
- 11) Оценка результатов рефакторинга.
- 12) Особенности рефакторинга мобильных и веб-приложений.
- 13) Проблемы рефакторинга.

### **Промежуточная аттестация по итогам освоения дисциплины (экзамен).**

Предполагает ответ студента на два теоретических и один практический вопрос. Теоретические вопросы сгруппированы в комплект билетов, который формируется на основе статистики посещения занятий. Для стимулирования самостоятельной работы билеты формируются так, чтобы более высокой была доля вопросов, рассмотренных на занятиях с меньшей посещаемостью.

Практический вопрос выдвигается студенту после ответа на теоретические вопросы и предполагает решение задачи на компьютере. Задача выбирается на основе технологии уровневой дифференциации из базы задач, имеющейся у преподавателя. Во время решения задачи студент может пользоваться справочной литературой и конспектами.

### **Примерный перечень вопросов к экзамену**

- 1) Определение рефакторинга. Причины необходимости в рефакторинге. Цели рефакторинга. Роль рефакторинга в жизненном цикле разработки программного обеспечения.
- 2) Понятие «запахов кода». Признаки дурно пахнущего кода.
- 3) Технический долг. Причины появления «технического долга».
- 4) Запахи кода. Раздувальщики кода.
- 5) Запахи кода. Нарушители объектного дизайна.
- 6) Запахи кода. Утяжелители изменений.
- 7) Запахи кода. Замусориватели.
- 8) Запахи кода. Опутыватели связями.
- 9) Инструментальные средства рефакторинга.
- 10) Классификация шаблонов проектирования.
- 11) Шаблон Наблюдатель. Примеры использования.
- 12) Шаблон Декоратор. Примеры использования.
- 13) Шаблон Адаптер. Примеры использования.
- 14) Шаблон Фабрика. Примеры использования.
- 15) Шаблон Абстрактная фабрика. Примеры использования.
- 16) Шаблон Команда. Примеры использования.
- 17) Шаблон Состояние. Примеры использования.
- 18) Шаблон Коллекция идентичности объектов.
- 19) Шаблон Преобразователь данных.
- 20) Шаблон Загрузка по требованию.
- 21) Шаблон Единица работы.
- 22) План мероприятий по организации рефакторинга. Проверочный лист рефакторинга.
- 23) Техники рефакторинга. Составление методов.
- 24) Техники рефакторинга. Перемещение функций между объектами.
- 25) Техники рефакторинга. Организация данных.
- 26) Техники рефакторинга. Упрощение условных выражений.
- 27) Техники рефакторинга. Упрощение вызовов методов.
- 28) Техники рефакторинга. Решение задач обобщения.



29) Поддержание единства проекта при рефакторинге. Документирование рефакторинга. Оценка результатов рефакторинга.

30) Особенности рефакторинга мобильных и веб-приложений. Проблемы рефакторинга.

**Самостоятельная работа** студентов по дисциплине «Рефакторинг программного кода» включает в себя следующие виды деятельности:

- 1) освоение материалов, слабо освещённых в рамках лекционного курса;
- 2) выполнение домашних заданий на основе материала, рассмотренного на практических занятиях;
- 3) подготовку по всем видам контрольных мероприятий, в том числе к текущему контролю знаний и промежуточной аттестации.

Задания предполагают написание работоспособного программного кода, реализующего методы рефакторинга, освещённые в лекционном материале и на практических занятиях. Проверка выполнения заданий осуществляется преподавателем в дистанционном режиме (по электронной почте или с использованием иных средств коммуникации).

#### Типовые задания для самостоятельной работы студентов

- 1) Практическая реализация шаблона Коллекция идентичности объектов.
- 2) Практическая реализация шаблона Преобразователь данных.
- 3) Практическая реализация шаблона Загрузка по требованию.
- 4) Практическая реализация шаблона Единица работы.
- 5) Практическая реализация шаблона Адаптер.
- 6) Практическая реализация шаблона Состояние.
- 7) Практическая реализация шаблона Абстрактная фабрика.
- 8) Практическая реализация техники рефакторинга «Составление методов».
- 9) Практическая реализация техники рефакторинга «Перемещение функций между объектами».
- 10) Практическая реализация техники рефакторинга «Организация данных».
- 11) Практическая реализация техники рефакторинга «Упрощение условных выражений».
- 12) Практическая реализация техники рефакторинга «Упрощение вызовов методов».
- 13) Практическая реализация техники рефакторинга «Решение задач обобщения».

Фонд оценочных средств для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 7.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ	
		Количество экземпляров в издании в библиотеке ВлГУ в соответствии с ФГОС ВО	Наличие в электронной библиотеке ВлГУ
1	2	3	4
Основная литература			
1. Практическая программная инженерия на основе учебного примера: Учебное пособие / Мацяшек Л.А., Лионг Б.Л., - 3-е изд., (эл.) - М.:БИНОМ. Лаб. знаний, 2015. - 959 с.: ISBN 978-5-9963-2499-6	2015		<a href="http://www.studentlibrary.ru/book/ISBN9785996311828.html">http://www.studentlibrary.ru/book/ISBN9785996311828.html</a>
2. Архитектура и проектирование программных систем : монография / С.В. Назаров. — 2-е изд., перераб. и доп. — М. : ИНФРА-М, 2018. — 374 с.	2018		<a href="http://znanium.com/catalog/product/925839">http://znanium.com/catalog/product/925839</a>
3. Введение в программную инженерию : Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчев. — М.: КУРС: ИНФРА-М, 2019. — 336 с. – ISBN 978-5-906923-22-6	2019		<a href="http://znanium.com/catalog/product/1035160">http://znanium.com/catalog/product/1035160</a>
Дополнительная литература			
1. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования [Электронный ресурс] / Э. Гамма, Р. Хелм,	2007		<a href="http://znanium.com/catalog/product/407366">http://znanium.com/catalog/product/407366</a>

Р. Джонсон, Д. Влиссидес; Пер. с англ. - Москва : ДМК Пресс, 2007. - 368 с.: ил. - (Серия «Для программистов»). - ISBN 5-93700-023-4			
2. Архитектура корпоративных информационных систем/Астапчук В.А., Терещенко П.В. - Новосиб.: НГТУ, 2015. - 75 с.: ISBN 978-5-7782-2698-2	2015		<a href="http://znanium.com/bookread2.php?book=546624">http://znanium.com/bookread2.php?book=546624</a>
3. Самоучитель UML: Самоучитель / Леоненков А.В., - 2-е изд., перераб. и доп. - СПб:БХВ-Петербург, 2015. - 418 с. ISBN 978-5-9775-1216-9	2015		<a href="http://znanium.com/catalog/product/939591">http://znanium.com/catalog/product/939591</a>

### 7.2. Периодические издания

1. Computerworld Россия, ISSN: 1560-5213.
2. Мир ПК, ISSN: 0235-3520.

### 7.3. Интернет-ресурсы

1. Рефакторинг.Гуру // Режим доступа: <https://refactoring.guru/ru>
2. Refactoring.com. Сайт Мартина Фоулера о рефакторинге. // Режим доступа: <https://www.refactoring.com/>
3. Паттерны проектирования // Режим доступа: <http://cpp-reference.ru/patterns/>
4. Ольга Дубина. Обзор паттернов проектирования // Режим доступа: <http://citforum.ru/SE/project/pattern/>
5. Develop effective XML documents using structural design patterns. // Режим доступ: <http://www.xmlpatterns.com/>

## 8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации данной дисциплины имеются специальные помещения для проведения занятий лекционного типа, занятий практического типа, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Практические занятия проводятся в аудитории (компьютерном классе) 511г-3 (или аналогичном компьютерном классе в зависимости от сетки расписания).

Перечень используемого лицензионного программного обеспечения:

- 1) MS Word;
- 2) MS Visual Studio;
- 3) Eclipse Juno (свободная лицензия Eclipse Public License).



Рабочую программу составил Лексин А.Ю. \_\_\_\_\_  
(ФИО, подпись)

Рецензент  
(представитель работодателя) Ген. директор ООО "ФС Сервис" Квасов Ф.С. \_\_\_\_\_  
(место работы, должность, ФИО, подпись)

Программа рассмотрена и одобрена на заседании кафедры физики и прикладной математики  
Протокол № 1 от 02.09.2019 года  
Заведующий кафедрой \_\_\_\_\_ Аракелян С.М.  
(ФИО, подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии  
направления 02.03.02 Фундаментальная информатика и информационные технологии  
Протокол № 1 от 02.09.2019 года  
Председатель комиссии \_\_\_\_\_ Аракелян С.М.  
(ФИО, подпись)

**ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ  
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ**

Рабочая программа одобрена на 2020-2021 учебный год

Протокол заседания кафедры № 1 от 31.08.2020 года

Заведующий кафедрой \_\_\_\_\_ С. М. Архипов

Рабочая программа одобрена на \_\_\_\_\_ учебный год

Протокол заседания кафедры № \_\_\_\_\_ от \_\_\_\_\_ года

Заведующий кафедрой \_\_\_\_\_

Рабочая программа одобрена на \_\_\_\_\_ учебный год

Протокол заседания кафедры № \_\_\_\_\_ от \_\_\_\_\_ года

Заведующий кафедрой \_\_\_\_\_

