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)**

Институт прикладной математики, физики и информатики

УТВЕРЖДАЮ:
Директор института
_____ К.С. Хорьков
«30» 08 2021 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
РЕФАКТОРИНГ ПРОГРАММНОГО КОДА

направление подготовки / специальность

02.03.02 Фундаментальная информатика и информационные технологии
(код и наименование направления подготовки (специальности)

направленность (профиль) подготовки

Мобильные и Интернет-технологии
(направленность (профиль) подготовки))

г. Владимир

2021

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины: знакомство с теоретическими основами и методиками рефакторинга программного кода.
Для достижения цели предполагается решение следующих задач:

- 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

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

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции *(код, содержание индикатора*	Результаты обучения по дисциплине	
ПК-1. Способен разрабатывать требования и проектировать программное обеспечение	ПК-1.1. Знает методики разработки требований к системе, методы классического системного анализа, стандарты оформления технических заданий, методологии разработки программного обеспечения и технологии программирования, языки формализации функциональных спецификаций, принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения, основные концепции и атрибуты качества программного обеспечения. ПК-1.2. Умеет выбирать методики разработки требований к системе и шаблоны документов требований к системе, определять источники информации для требований к системе, выполнять тестирование системы с целью проверки её реализации на соответствие требованиям, формулировать и оформлять запросы на изменение требований, проводить анализ исполнения требований, вырабатывать варианты их реализации, проводить оценку и обоснование рекомендуемых	Знает: • методы и средства проектирования программного обеспечения; • язык формализации функциональных спецификаций UML; • принципы построения архитектуры программного обеспечения на базе шаблонов (паттернов) проектирования и методов рефакторинга; • основные концепции и атрибуты качества программного обеспечения. Умеет: • вырабатывать варианты реализации требований с использованием методов рефакторинга; • проводить оценку и обоснование рекомендуемых решений; • выбирать средства и варианты рефакторинга программного обеспечения; • использовать существующие типовые решения и шаблоны проектирования программного обеспечения; • применять методы и средства	Контрольные вопросы к текущей и промежуточной аттестации.

	решений, выбирать и использовать средства и варианты реализации программного обеспечения. ПК-1.3. Владеть навыками планирования работ по разработке требований к системе, анализа проблемной ситуации, согласования целей создания системы с заинтересованными лицами, оформления технического задания на систему, представления концепции, технического задания и изменений в них заинтересованным лицам, оценки возможностей, времени и трудоемкости реализации требований к программному обеспечению, разработки и согласования технических спецификаций на программное обеспечение, формирования и предоставления отчётности в соответствии с установленными регламентами, проектирования структур данных, баз данных, программных интерфейсов.	рефакторинга программного кода; • оценивать соответствие требованиям существующих систем и их аналогов. Владеет: • навыками оценки возможностей, времени и трудоемкости реализации требований к программному обеспечению; • навыками проектирования структур данных, программных интерфейсов; • навыками выбора стиля написания кода; • навыками анализа проблемной ситуации.	
ПК-3. Способен оценивать, выбирать и реализовывать варианты архитектуры мобильных и веб-приложений	ПК-3.1. Знает методы и технологии анализа, проектирования и разработки мобильных и веб-приложений, шаблоны проектирования слоёв или компонентов программного обеспечения, принципы обеспечения отказоустойчивости программных компонентов, принципы информационной безопасности. ПК-3.2. Умеет декомпозировать программные средства на компоненты, определять программный интерфейс компонентов, применять шаблоны проектирования при реализации мобильных и веб-приложений, оценивать риски с точки зрения информационной безопасности. ПК-3.3. Владеет навыками оценки, выбора и проектирования однослойной и многослойной архитектуры мобильных и веб-приложений, выбора механизмов авторизации, аутентификации и поддержки сеанса, оценки и выбора технологии доступа к данным, анализа качества кода, реализации мобильных и веб-приложений на уровне модулей и системы в целом.	Знает: • шаблоны проектирования слоёв или компонентов программного обеспечения; • стили написания кода; • методы и технологии анализа, проектирования и разработки мобильных и веб-приложений на стадии рефакторинга программного кода. Умеет: • декомпозировать программные средства на компоненты; • применять шаблоны проектирования при реализации мобильных и веб-приложений; • определять программный интерфейс компонентов. Владеет: • навыками оценки, выбора и проектирования однослойной и многослойной архитектуры мобильных и веб-приложений; • навыками выбора стиля написания кода.	Контрольные вопросы к текущей и промежуточной аттестации.

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 3 зачетных единицы, 108 часов.

Тематический план
форма обучения – очная (ускоренное обучение)

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Контактная работа обучающихся с педагогическим работником				Самостоятельная работа	Формы текущего контроля успеваемости, форма промежуточной аттестации *(по семестрам)*
				Лекции	Практические занятия	Лабораторные работы	в форме практической подготовки		
1	Общие принципы и инструментальные средства рефакторинга программного кода.	3	1-6	4	6	–	8	13	рейтинг-контроль №1
2	Шаблоны проектирования и рефакторинга	3	5-14	8	8	–	10	18	
3	Организация и оценка результатов рефакторинга	3	13-18	6	4	–	6	14	рейтинг-контроль №2
Всего за 4 семестр:		–	–	18	18	–		45	экзамен (27)
Наличие в дисциплине КП/КР		–	–	–	–	–	–	–	–
Итого по дисциплине		–	–	18	18	–	–	45	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. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

5.1. Текущий контроль успеваемости

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.*

5.2. 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 выдаётся 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.*

5.3. Самостоятельная работа обучающегося.

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 материалов (ФОМ) для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ Наличие в электронном каталоге ЭБС
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	http://www.studentlibrary.ru/book/ISBN9785996311828.html
2. Архитектура и проектирование программных систем : монография / С.В. Назаров. — 2-е изд., перераб. и доп. — М. : ИНФРА-М, 2018. — 374 с.	2018	http://znanium.com/catalog/product/925839
3. Введение в программную инженерию : Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. — М.: КУРС: ИНФРА-М, 2019. — 336 с. – ISBN 978-5-906923-22-6	2019	http://znanium.com/catalog/product/1035160
Дополнительная литература		
1. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования [Электронный ресурс] / Э. Гамма, Р. Хелм, Р. Джонсон, Д.	2007	http://znanium.com/catalog/product/40736

Влиссидес; Пер. с англ. - Москва : ДМК Пресс, 2007. - 368 с.: ил. - (Серия «Для программистов»). - ISBN 5-93700-023-4		6
2. Архитектура корпоративных информационных систем/АстапчукВ.А., ТерещенкоП.В. - Новосиб.: НГТУ, 2015. - 75 с.: ISBN 978-5-7782-2698-2	2015	http://znanium.com/bookread2.php?book=546624
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	http://znanium.com/catalog/product/939591

6.2. Периодические издания

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

6.3. 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/

7. 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 доц. каф. ФиПМ Лексин А.Ю. _____
(ФИО, должность, подпись)

Рецензент
(представитель работодателя)
Генеральный директор ООО «ФС Сервис» _____ Д.С. Квасов
(место работы, должность, ФИО, подпись)

Программа рассмотрена и одобрена на заседании кафедры физики и прикладной математики
Протокол № 1 от 30.08.2021 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 30.08.2021 года
Председатель комиссии _____ _____Аракелян С.М.
(ФИО, подпись)

ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой_____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой_____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой_____