

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

И.о. заведующего кафедрой
ПОиАИС

 Барановский Е.С.

27.03.2025г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.В.ДВ.03.03 Программирование для мобильных устройств

- 1. Шифр и наименование направления подготовки:** 02.03.03
Математическое обеспечение и администрирование информационных систем
- 2. Профиль подготовки (при наличии):**
Проектирование и разработка информационных систем
- 3. Квалификация (степень) выпускника:**
бакалавр
- 4. Форма обучения:**
очная
- 5. Кафедра, отвечающая за реализацию дисциплины:**
Программного обеспечения и администрирования информационных систем
- 6. Составители программы:**
Меджидов Р.Г.
- 7. Рекомендована:**
НМС факультета Прикладной математики информатики и механики №6 от 17.03.2025 г.
- 8. Учебный год:** 2028-2029

Семестр(ы): 7

9. Цели и задачи дисциплины

Целями освоения дисциплины являются:

- приобретение знаний и навыков в области проектирования приложений, рассчитанных на длительную поддержку и возможность расширения.,
- приобретение опыта создания автоматических тестов.

Задачи дисциплины:

- ознакомление с понятием проектирования приложений,
- изучение типичных проблем проектирования,
- изучение принципов SOLID, их достоинств и недостатков, критериев их применимости в различных ситуациях,
- изучение возможностей практического применения принципов SOLID,
- изучение автоматических тестов,
- ознакомление с подходом Test-driven development.

10. Место дисциплины в структуре ООП:

Является дисциплиной по выбору для программы бакалавриата.

11. Планируемые результаты обучения по дисциплине (знания, умения, навыки), соотнесенные с планируемыми результатами освоения программы (компетенциями):

Код	Название компетенции	Коды	Индикаторы	Планируемые результаты обучения
ПКВ-4	Способен проектировать и верифицировать архитектуру, а также разрабатывать прототипы информационных систем			Знать: признаки проблем проектирования и способы их разрешения; принципы SOLID, их достоинства и недостатки; виды тестов; юнит-тесты; TDD. Уметь: избегать ошибок проектирования приложения; применять принципы SOLID; обосновать целесообразность применения того или иного принципа для данной ситуации; разрабатывать автоматические тесты. Владеть: навыками создания приложений с использованием принципов SOLID и TDD.

12. Объем дисциплины в зачетных единицах/час. — 2/72.

Форма промежуточной аттестации зачет с оценкой.

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость	
		Всего	По семестрам
			Семестр 7
Аудиторные занятия		32	32
в том числе:	лекции	0	0
	практические	0	0
	лабораторные	32	32
Самостоятельная работа		40	40
Форма промежуточной аттестации (экзамен, зачет – час.)			
Итого:		72	72

13.1. Содержание дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Онлайн-курс, ЭУК*
3. Лабораторные занятия			
3.1	Понятие дизайна (проектирования) приложения. Признаки проблемного дизайна.	Понятие дизайна (проектирования) приложения. Признаки проблемного дизайна: жесткость, неподвижность, вязкость, ненужная сложность, ненужная повторяемость, плохая читабельность, хрупкость.	Онлайн-курс «Принципы гибкой разработки ПО (ПОиАИС)» (https://edu.vsu.ru/course/view.php?id=27341)
3.2	Принципы SOLID.	Принцип единственной ответственности. Принцип открытости-закрытости. Принцип подстановки Лисков. Принцип разделения интерфейса. Принцип инверсии зависимости.	
3.3	Тестирование. UT. TDD.	Виды тестов. Юнит-тесты. Понятие Test-driven development.	

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1	Понятие дизайна (проектирования) приложения. Признаки проблемного дизайна.			5	6	11
2	Принципы SOLID.			22	28	50
3	Тестирование. UT. TDD.			5	6	11
	Итого:			32	40	72

14. Методические указания по освоению дисциплины:

Изучение материала, излагаемого на занятиях. Изучение предлагаемых на электронном курсе материалов. Выполнение практических заданий для самостоятельной работы, выполнение лабораторных работ по дисциплине. Рекомендуются вести конспект. Изучение рекомендованной литературы и методических материалов.

15. Перечень литературы, ресурсов интернет, необходимых для освоения дисциплины

а) основная литература:

№ п/п	Источник
1	Принципы гибкой разработки программного обеспечения / Артемов М.А., Золотарев С.В., Барановский Е.С., Курбатова И.В. — Воронеж : Издательский дом ВГУ, 2018 — 48 с.

б) дополнительная литература:

№ п/п	Источник
1	Принципы Solid и как они помогают сделать код лучше – https://habr.com/ru/companies/itentika/articles/694730/ (дата обращения: 01.04.2024).
2	Принципы SOLID, только понятно – https://habr.com/ru/articles/811305/ (дата обращения: 01.04.2024).
3	SPR (Single responsibility principle) - Принцип единственной обязанности – https://makedev.org/principles/solid/srp.html (дата обращения: 01.04.2024).
4	Принцип открытости-закрытости – https://habr.com/ru/companies/tbank/articles/472186/ (дата обращения: 01.04.2024).
5	Принцип открытости/закрытости – https://blog.byndyu.ru/2009/10/blog-post_14.html (дата обращения: 01.04.2024).
6	SOLID в деталях: принцип открытости/закрытости – https://alexkosarev.name/2023/02/13/solid-in-details-оср/ (дата обращения: 01.04.2024).
7	Принцип подстановки Лисков – https://habr.com/ru/companies/tbank/articles/490738/ (дата обращения: 01.04.2024).
8	SOLID в деталях: Принцип подстановки Лисков – https://alexkosarev.name/2023/03/02/solid-in-details-lsp/ (дата обращения: 01.04.2024).
9	Принцип подстановки Барбары Лисков – https://habr.com/ru/articles/739734/ (дата обращения: 01.04.2024).
10	LSP. Введение – https://solidbook.vercel.app/lsp (дата обращения: 01.04.2024).
11	Разбираемся с SOLID: Инверсия зависимостей – https://habr.com/ru/articles/313796/ (дата обращения: 01.04.2024).
12	Пошаговое руководство. Разработка на основе тестов с помощью тестовой Обозреватель – https://learn.microsoft.com/ru-ru/visualstudio/test/quick-start-test-driven-development-with-test-explorer?view=vs-2022 (дата обращения: 01.04.2024).
13	Пошаговое руководство. Создание и запуск модульных тестов для управляемого кода – https://learn.microsoft.com/ru-ru/visualstudio/test/walkthrough-creating-and-running-unit-tests-for-managed-code?view=vs-2022 (дата обращения: 01.04.2024).

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

№ п/п	Ресурс
1	www.lib.vsu.ru – ЗНБ ВГУ
2	Онлайн-курс «Принципы гибкой разработки ПО (ПОиАИС)» (https://edu.vsu.ru/course/view.php?id=27341)

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1	ОС Windows
2	MS Office
3	Visual Studio 2012 и выше

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ, электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины используются модульно-рейтинговая и личностно-ориентированные технологии обучения (ориентированные на индивидуальность студента, компьютерные и коммуникационные технологии). Дисциплина реализуется с применением электронного обучения и дистанционных образовательных технологий, для организации самостоятельной работы обучающихся используется онлайн-курс, размещенный на платформе Электронного университета ВГУ (LMS moodle), а также другие Интернет-ресурсы, приведенные в п.15б и п.15в.

18. Материально-техническое обеспечение дисциплины:

Лабораторные занятия должны проводиться в специализированной аудитории, оснащенной учебной мебелью и персональными компьютерами с доступом в сеть Интернет (компьютерные классы, студии), мультимедийным оборудованием (мультимедийный проектор, экран, средства звуковоспроизведения), Число рабочих мест в аудитории должно быть таким, чтобы обеспечивалась индивидуальная работа студента на отдельном персональном компьютере.

Для самостоятельной работы необходим компьютер с доступом к сети Интернет.

Программное обеспечение: Visual Studio 2012 и выше.

19. Оценочные средства для проведения текущего контроля и промежуточной аттестаций

19.1. Текущий контроль

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

лабораторные работы, контрольная работа

Пример лабораторной работы 1

Написать программу, демонстрирующую нарушение принципа «SRP» по тематике «Грузовые суда и причалы». Затем продемонстрировать ту же задачу с соблюдением принципа. Язык выполнения работы — C#.

Пример лабораторной работы 2

Написать программу, демонстрирующую 3 нарушения проектирования приложения тематике «Медицина». Показать что нужно исправить для создания правильного дизайна приложения. Язык выполнения работы — C#.

Пример лабораторной работы 3

Написать программу, содержащую набор тестов для поставленной задачи. Тема — «Космос и научная фантастика». Создать класс с методами, которые пройдут все созданные тесты без ошибок. Язык выполнения работы — C#.

Описание технологии проведения

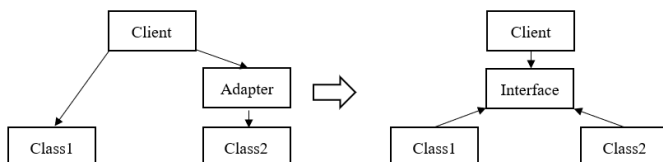
Выполнение лабораторной работы. Демонстрация ее кода и работы. Пояснения к выполнению работы. Описание алгоритма. Устный ответ на вопросы.

Требования к выполнению заданий (или шкалы и критерии оценивания)

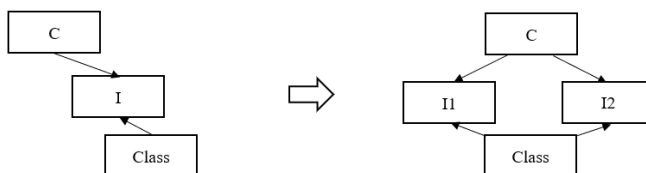
Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Лабораторная работа выполнена корректно, не содержит необработанных исключений, поставленная цель достигнута, алгоритм решения оптимальный. Свободно отвечает на все вопросы; в случае незнания небольшой части материала способен выстроить собственную логическую цепочку рассуждений и получить ответ.	базовый	Принято
Лабораторная работа содержит ошибки, имеются ситуации, приводящие к падению задачи, выбран неподходящий алгоритм решения, цель задачи не достигнута или достигнута частично. Не отвечает или отвечает не на все вопросы.	—	Не принято

Пример контрольной работы

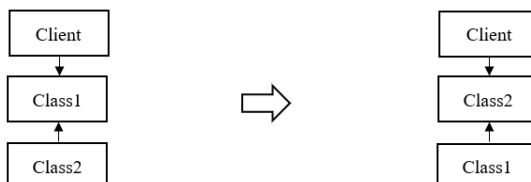
1. Явный признак нарушения SRP:
 - a) антипаттерн «божественный объект»;
 - b) паттерн «стратегия»;
 - c) MRP;
 - d) консольный ввод-вывод.
2. Указать порядок действий при TDD:
 - ___ формирование требований,
 - ___ написание юнит-тестов,
 - ___ реализация основного функционала.
3. Какому из принципов SOLID соответствует схема?



4. Какому из принципов SOLID соответствует схема?



5. Какому из принципов SOLID соответствует схема?



Требования к выполнению заданий (или шкалы и критерии оценивания)

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Дано 80% или более правильных ответов.	базовый	Принято
Дано 79% или менее правильных ответов.	—	Не принято

19.2. Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

опрос

Описание технологии проведения

Успешная сдача всех лабораторных и контрольных работ. Ответ на устный вопрос, демонстрирующий теоретические знания.

Требования к выполнению заданий, шкалы и критерии оценивания

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Сданы все 7 лабораторных работ. Контрольная написана успешно. При устном ответе: свободно владеет материалом; в случае незнания небольшой части материала способен выстроить собственную логическую цепочку рассуждений и получить ответ, усвоение основной литературы, умеет приводить примеры.	Повышенный	Отлично
Сданы 6 из 7 лабораторных работ. Контрольная написана успешно. При устном ответе: полное знание учебно-программного материала, усвоение основной литературы, умеет приводить примеры.	Базовый	Хорошо
Сданы 5 из 7 лабораторных работ. Контрольная написана успешно. При устном ответе: знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, знакомство с основной литературой, рекомендованной программой.	Пороговый	Удовлетворительно
Сдано меньше, чем 5 лабораторных работ или не сдана контрольная работа. При устном ответе имеются пробелы в знаниях основного материала, наличие которых препятствует дальнейшему обучению студента, не способен приводить примеры.	—	Неудовлетворительно