

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ

Декан факультета ФИТКБ



Бредихин А.В./

28.08.2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

**«Разработка прикладного программного обеспечения для
компьютерных систем (связь, информационные и
коммуникационные технологии)»**

Специальность 10.05.01 Компьютерная безопасность

Специализация специализация № 4 "Безопасность компьютерных систем и
сетей (связь, информационные и коммуникационные технологии)"

Квалификация выпускника специалист по защите информации

Нормативный период обучения 5 лет и 6 м.

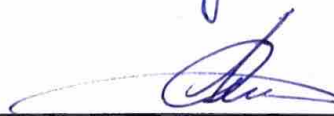
Форма обучения очная

Год начала подготовки 2025

Автор программы

 _____ Е.А. Москалева

Заведующий кафедрой
Систем информационной
безопасности

 _____ А.Г. Остапенко

Руководитель ОПОП

 _____ С.С. Куликов

Воронеж 2025

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Целью дисциплины является приобретение студентами знаний об особенностях разработки компонент программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности, современных средах разработки и тестирования программного обеспечения, стандартов по разработке документации на программное обеспечение.

1.2. Задачи освоения дисциплины

- сформировать у будущего специалиста компетенции направленные на способность разработки компонент программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности;
- способность разрабатывать и анализировать корректность политики информационной безопасности компьютерных систем и сетей (связь, информационные и коммуникационные технологии)
- проводить разработку программного обеспечения в соответствии с существующими технологиями промышленной разработки программных средств

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Разработка прикладного программного обеспечения для компьютерных систем (связь, информационные и коммуникационные технологии)» относится к дисциплинам обязательной части блока Б1.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Разработка прикладного программного обеспечения для компьютерных систем (связь, информационные и коммуникационные технологии)» направлен на формирование следующих компетенций:

ОПК-13 - Способен разрабатывать компоненты программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности;

ОПК-4.3 - Способен разрабатывать и анализировать корректность политики информационной безопасности компьютерных систем и сетей (связь, информационные и коммуникационные технологии)

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-13	знать особенности разработки компонент программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности
	уметь разрабатывать компоненты программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности

	владеть средствами разработки программного обеспечения в том числе средств повышения эффективности процессов разработки и эксплуатации безопасного программного обеспечения за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации
ОПК-4.3	знать цели и задачи процедур анализа корректности политик безопасности, в том числе с применением разработанного прикладного программного обеспечения для компьютерных систем
	уметь проводить анализ корректности политик безопасности для компьютерных систем и сетей с применением прикладного программного обеспечения

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Разработка прикладного программного обеспечения для компьютерных систем (связь, информационные и коммуникационные технологии)» составляет 8 з.е.

Распределение трудоемкости дисциплины по видам занятий
очная форма обучения

Виды учебной работы	Всего часов	Семестры	
		8	9
Аудиторные занятия (всего)	198	108	90
В том числе:			
Лекции	108	54	54
Лабораторные работы (ЛР)	90	54	36
Самостоятельная работа	54	36	18
Курсовой проект	+		+
Часы на контроль	36		36
Виды промежуточной аттестации - зачет с оценкой, экзамен	+	+	+
Общая трудоемкость:			
академические часы	288	144	144
зач.ед.	8	4	4

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Паттерны проектирования программного обеспечения	Введение в паттерны проектирования. Проектирование редактора документов. Порождающие паттерны. Структурные паттерны. Паттерны поведения.	20	18	10	48
2	Распределённое и параллельное программирование	Программирование с разделяемыми переменными: процессы и синхронизация; блокировки и барьеры; семафоры; мониторы; реализация	34	24	12	70

		Распределенное программирование: передача сообщений; удаленный вызов процедур и рандеву; модели взаимодействия процессов; реализация языковых механизмов Синхронное параллельное программирование: научные вычисления, языки, компиляторы, библиотеки и инструментальные средства				
4	Высоконагруженные приложения	Основы разработки: подходы, надёжность, масштабируемость, сопровождение. Модели данных и языки запросов. Подсистемы хранения и извлечения данных. Распределённые данные: ведущие и ведомые узлы, задержка репликации, репликация с несколькими ведущими узлами, репликация без ведущего узла. Секционирование. Транзакции. Согласованность и консенсус. Пакетная и потоковая обработки. Вопрос безопасности	18	18	10	46
5	Микросервисы	Введение в микросервисную архитектуру. Стратегии декомпозиции. Межпроцессное взаимодействие в микросервисной архитектуре. Управление транзакциями с помощью повествований. Проектирование бизнес-логики в микросервисной архитектуре. Разработка бизнес-логики с порождением событий. Реализация запросов в микросервисной архитектуре. Шаблоны внешних API. Тестирование микросервисов. Разработка сервисов, готовых к промышленному использованию. Развертывание микросервисов. Процесс перехода на микросервисы.	18	18	10	46
6	Безопасный DevOps	Основы безопасности в DevOps. Выстраивание базового DevOps-конвейера: защита веб-приложений, защита облачной инфраструктуры, защита каналов взаимодействия, защита конвейера поставки. Выявление аномалий и защита сервисов от атак. Сбор и хранение журналов. Анализ журналов для выявления вторжений и атак. Обнаружение вторжений. Практические примеры реагирования на инцидент	18	12	12	42
Итого			108	90	54	252

5.2 Перечень лабораторных работ

Разработка программного проекта с использованием паттернов проектирования на C++ в среде Visual Studio.

Параллельное программирование в стандарте MPI.

Введение в микросервисную архитектуру. Получение базовых навыков по работе с Docker.

DevOps на сервисах Amazon AWS.

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 7 семестре для очной формы обучения.

Примерная тематика курсового проекта: «Разработка автоматизированного клиент-серверного WEB-ориентированного приложения»

Задачи, решаемые при выполнении курсового проекта:

- исследование и реализация требований к программному обеспечению серверной части;

- исследование и реализация требований к клиентскому программному обеспечению

- рассмотрение вопросов безопасности web-приложения

Курсовой проект включает в себя графическую часть и расчетно-пояснительную записку.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ОПК-13	знать особенности разработки компонент программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности	знание особенностей разработки компонент программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь разрабатывать компоненты про-	умение разрабатывать компоненты программ-	Выполнение работ в срок,	Невыполнение работ в срок,

	граммных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности	ных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности	предусмотренный в рабочих программах	предусмотренный в рабочих программах
	владеть средствами разработки программного обеспечения в том числе средств повышения эффективности процессов разработки и эксплуатации безопасного программного обеспечения за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации	владение средствами разработки программного обеспечения в том числе средств повышения эффективности процессов разработки и эксплуатации безопасного программного обеспечения за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-4.3	знать цели и задачи процедур анализа корректности политик безопасности, в том числе с применением разработанного прикладного программного обеспечения для компьютерных систем	знание цели и задачи процедур анализа корректности политик безопасности, в том числе с применением разработанного прикладного программного обеспечения для компьютерных систем	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь проводить анализ корректности политик безопасности для компьютерных систем и сетей с применением прикладного программного обеспечения	умение проводить анализ корректности политик безопасности для компьютерных систем и сетей с применением прикладного программного обеспечения	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 7, 8 семестре для очной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ОПК-13	знать особенности разработки компонент программных и программно-аппаратных средств защиты информации в компьютерных системах и	Тест	Выполнение теста на 90- 100%	Выполнение теста на 80- 90%	Выполнение теста на 70- 80%	В тесте менее 70% правильных ответов

	проводить анализ их безопасности					
	уметь разрабатывать компоненты программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности	Решение стандартных практически х задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть средствами разработки программного обеспечения в том числе средств повышения эффективности процессов разработки и эксплуатации безопасного программного обеспечения за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ОПК-4.3	знать цели и задачи процедур анализа корректности политик безопасности, в том числе с применением разработанного прикладного программного обеспечения для компьютерных систем	Тест	Выполнение теста на 90- 100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь проводить анализ корректности политик безопасности для компьютерных систем и сетей с применением прикладного программного обеспечения	Решение стандартных практически х задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1.Важным начальным этапом при работе с паттернами является...
Описание окружение системы, с которой планируется осуществлять взаимодействие +

Создание "Простого" объекта без увеличения сложности структуры программного обеспечения

Адекватное моделирование рассматриваемой предметной области +
Организация использования функций определенного бизнес объекта,

недоступного для модификации

2. Использование паттернов проектирования на практике дает проектировщику следующие неоспоримые преимущества...

Модель системы, построенная в терминах паттернов проектирования, фактически является структурированным выделением тех элементов и связей, которые значимы при решении поставленной задачи +

Модель, построенная с использованием паттернов проектирования, более проста и наглядна в изучении. Несмотря на простоту и наглядность, она позволяет глубоко и всесторонне проработать архитектуру разрабатываемой системы +

Применение паттернов проектирования позволяет принимать решения, которые будут успешны для решения всех проблем

Применение паттернов проектирования повышает устойчивость системы к изменению требований и упрощает неизбежную последующую доработку +

3. Общеизвестны три фундаментальные и наиболее значимые аспекты объектно-ориентированного проектирования. Перечислите их.

Рекурсия

Полиморфизм +

Наследование +

Инкапсуляция +

4. Преимущество использования инкапсуляции состоит в том, что оно...

Упрощает разбиение программы +

Сохраняет целостность, прозрачность, предсказуемость создаваемого программного обеспечения

Упрощает добавление новых примитивных или составных объектов

Упрощает выявление и обработку исключительных ситуаций

5. Следует инкапсулировать те концепции, которые...

Постоянны

Имеют потомков

Могут изменяться +

Являются временными

6. Перечислите архитектурные стили.

"Наследование классов"

"Программа-сопрограмма" +

"Черный ящик"

"Конвейеры и фильтры" +

7. В качестве примера наиболее популярных антишаблонов проектирования можно привести следующие...

Случайная сложность +

Соккрытие ошибок +

Действие на расстояние +
Рефакторинг кода

8. Шаблон – это

Алгоритмы, основанные на возможности повторного использования преобразований и возможности модификации существующей системы посредством добавления новых преобразований

Единое унифицированное управление, основанное на структурных особенностях поэтапного выполнения программ

Общее решение некоторой повторяющейся проблемы в определенном контексте +

Экземпляр объекта, который выдает себя за группу самостоятельных экземпляров

9.Связующим звеном между архитектурой, стилем и шаблонами является...

Наследование классов

Инкапсуляция

Полиморфизм

Язык шаблонов +

10. Язык шаблонов, представляет собой...

Реализацию атрибутов и интеграционных компонентов конкретного программного обеспечения +

Коллекцию взаимодействующих между собой шаблонов, образующих в конечном итоге общее решение повторяющейся проблемы в определенном контексте +

Компонент с низкой степенью связанности

Последовательный доступ ко всем элементам составного объекта

7.2.2 Примерный перечень заданий для решения стандартных задач

1.Язык шаблонов состоит из ключевых компонентов...

Общее решение +

Определенный контекст +

Уникальная проблема

Повторяющаяся проблема +

2.Важность применения шаблонов для построения оптимальной архитектуры обусловлена следующими причинами...

Упрощение работы с существующей системой, если можно определить уникальный интерфейс взаимодействия

Однозначно определяются правила создания экземпляров

При использовании шаблонов в адекватном контексте вероятность получения успешно работающей физической реализации архитектуры возрастает +

Разработка и использование шаблонов в рамках компании обеспечивает

преимущества, связанные с их многократным использованием для решения различных проблем +

3. Сервис-ориентированная методология – это

Поведенческие и порождающие шаблоны проектирования +

Подход к проектированию прикладных информационных систем +

Визуальное изображение архитектуры +

Ключевой принцип, на основе которого реализуются механизмы, обеспечивающие гибкость, защиту системы от влияния изменений внешних систем

4. Сервис-ориентированная методология руководствуется следующими принципами

Явное отделение бизнес-логики от логики презентации информации +

Реализация бизнес-логики прикладной системы в виде некоторого количества компонентов или сервисов, которые доступны извне пользователям или другим компонентам +

Потребитель результатов, предоставляемых сервисом может быть прикладной системой или другим сервисом и имеет возможность вызвать оригинальный сервис самостоятельно +

Включение бизнес-логики в логику презентации информации

5. Основным недостатком шаблона "Абстрактной фабрики" является то что...

Непредвиденные обновления

Последующая возможная модернизация, подразумевающая включение новых объектов часто затруднительная и ресурсоемкая операция

Использование приводит к предсказуемому падению производительности

Протокол обновления не содержит никаких сведений о том, что изменилось в субъекте - работа наблюдателей при этом усложняется

6. Применение шаблона "абстрактной фабрики" позволяет получить следующие преимущества...

Гарантия использования только одного вида продуктов

Свободная замена семейства продуктов

Скрываются реализации конкретных классов

Всегда открыта реализация класса

7. Для того, чтобы реализация "Абстрактной фабрики" была выполнена необходимо следовать следующим рекомендациям...

Используя экземпляр объекта должен обращаться к абстрактной фабрике для создания требуемых экземпляров

Для каждого семейства создайте конкретные классы, производные от данного абстрактного класса

Однозначно идентифицируйте правила создания экземпляров

Упрощайте добавление новых операций к существующим в системе объектам

8. В качестве основных достоинств использования шаблона проектирования "Одиночка", можно выделить следующие...

Свободная замена семейства продуктов

Высокая гибкость при модификации класса

Контролируемый доступ к единственному экземпляру

Допускает уточнение операций над экземплярами и их представление

9. Основным недостатком паттерна "Одиночка" является...

Приводит к предсказуемому падению производительности

Непредвиденные обновления

Последующая возможная модернизация, подразумевающая включение новых объектов часто затруднительная и ресурсоемкая операция

В некоторых случаях приводит к созданию не масштабируемого приложения

10. Любая современная информационная система не должна зависеть от того, как в ней создаются, komponуются и представляются объекты.

Для того, чтобы практически поддержать данный постулат следует создавать новые объекты с помощью паттерна...

"Высокое зацепление"

"Хранитель"

"Контроллер"

"Прототип"

7.2.3 Примерный перечень заданий для решения прикладных задач

1. Отличительными условиями, характеризующими уместность применения шаблона "Прототип", являются...

Определение создаваемого объекта динамически, во время выполнения

Нежелательное создание отдельной иерархии классов фабрик для создания объектов-продуктов из параллельной иерархии классов (отличие от шаблона абстрактная фабрика)

Существует высокая сложность выявления и обработки исключительных ситуаций

Функционал клонирования объекта является более предпочтительным вариантом нежели его создание и инициализация с помощью конструктора

2. В условиях, когда не определено, какой из системных компонентов должен отвечать за создание экземпляров класса используют паттерн...

"Создатель экземпляров"

"Строитель"

"Прототип"

"Абстрактная фабрика"

3. Основным преимуществом использования паттерна "Создатель" экземпляров является...

Сохранение гибкости системы по причине того, что созданные объекты класса 1, как правило, видны только для класса 1

Архитектуру системы можно разрабатывать на базе объектов или их структур, созданных ранее

Все создаваемые объекты являются явно или потенциально применимыми для повторного использования

Система, построенная на принципах шаблона "Создатель экземпляров" значительно проще в понимании и дальнейшем сопровождении, особенно, если речь идет о сложных отраслях бизнеса

4. Если требуется отделить конструирование сложного объекта от его представления, таким образом, чтобы в результате одного и того же конструирования могли получаться различные представления используют шаблон...

"Создатель экземпляров"

"Строитель"

"Прототип"

"Абстрактная фабрика"

5. Суть шаблона "Строитель" состоит в том, что...

Алгоритм создания сложного объекта обязательно должен зависеть от того, из каких частей состоит объект и как эти части должны взаимодействовать между собой

Необходимо использовать формат данных, который должен распознаваться всеми преобразованиями

Алгоритм создания сложного объекта не должен зависеть от того, из каких частей состоит объект и как эти части должны взаимодействовать между собой

Алгоритмы, основанные на возможности повторного использования преобразований и возможности модификации существующей системы посредством добавления новых преобразований

6. В результате использования паттерна "Строитель" можно добиться следующих значимых преимуществ, укажите их.

Для изменения внутреннего представления "Продукта" достаточно определить новый вид "Строителя"

Простота расширения (добавление, изменение серверов, выполняющих обработку информации и их интеграцию с остальной частью системы)

Строитель предоставляет промежуточному объекту абстрактный интерфейс для конструирования необходимого продукта, за которым может скрыть представление, внутреннюю структуру и процесс сборки самого про-

дукта

Легкость добавления новых примитивных или составных объектов

7. В условиях, когда необходимо определить интерфейс для создания конкретного объекта, но требуется делегировать системным подклассам решение о том, какой класс нужно инстанцировать, используется шаблон...

"Контроллер"

"Фабричный метод"

"Прототип"

"Абстрактная фабрика"

8. Преимуществом шаблона "Фабричный метод" является то, что нет необходимости...

Встраивать в код зависящие от приложения классы

Объединять родственные операции в едином классе

Скрывать опасные/лишние данные

Заботится о производительности

9. Визуальное представление архитектуры имеет следующие достоинства...

Визуальное изображение архитектуры отображает аспекты взаимодействия различных сервисов с точки зрения реализации бизнес логики

Визуальное изображение архитектуры представляет собой общую площадку для обсуждения между сотрудниками, заинтересованными в ее разработке и тех, кто занимается ее непосредственным созданием +

Визуальное изображение архитектуры, понятное и согласованное всеми необходимыми сотрудниками позволяет охватить полную картину создаваемого программного обеспечения, не упустив значимых деталей +

Адекватное моделирование рассматриваемой предметной области

10. Диаграммы классов создаются при логическом проектировании информационных систем. Они служат, для...

Моделирования данных. Анализ предметной области позволяет выявить основные характерные для нее сущности и связи между ними +

Для моделирования логики программных компонент +

Отображения поведения объектов, изображенных на ней классов

Для представления архитектуры. Можно выделить архитектурно значимые классы и показать их на диаграммах, описывающих архитектуру +

7.2.4 Примерный перечень вопросов для подготовки к зачету

Не предусмотрено учебным планом

7.2.5 Примерный перечень заданий для решения прикладных задач

Введение в паттерны проектирования. Проектирование редактора документов. Порождающие паттерны. Структурные паттерны. Паттерны поведения.

Программирование с разделяемыми переменными: процессы и синхронизация; блокировки и барьеры; семафоры; мониторы; реализация

Распределенное программирование: передача сообщений; удаленный вызов процедур и рандеву; модели взаимодействия процессов; реализация языковых механизмов

Синхронное параллельное программирование: научные вычисления, языки, компиляторы, библиотеки и инструментальные средства

Основы разработки: подходы, надёжность, масштабируемость, сопровождение. Модели данных и языки запросов. Подсистемы хранения и извлечения данных. Распределённые данные: ведущие и ведомые узлы, задержка репликации, репликация с несколькими ведущими узлами, репликация без ведущего узла. Секционирование. Транзакции. Согласованность и консенсус. Пакетная и потоковая обработки. Вопрос безопасности

Введение в микросервисную архитектуру. Стратегии декомпозиции. Межпроцессное взаимодействие в микросервисной архитектуре. Управление транзакциями с помощью повествований. Проектирование бизнес-логики в микросервисной архитектуре. Разработка бизнес-логики с порождением событий. Реализация запросов в микросервисной архитектуре. Шаблоны внешних API. Тестирование микросервисов. Разработка сервисов, готовых к промышленному использованию. Развертывание микросервисов. Процесс перехода на микросервисы.

Основы безопасности в DevOps. Выстраивание базового DevOps-конвейера: защита веб-приложений, защита облачной инфраструктуры, защита каналов взаимодействия, защита конвейера поставки. Выявление аномалий и защита сервисов от атак. Сбор и хранение журналов. Анализ журналов для выявления вторжений и атак. Обнаружение вторжений. Практические примеры реагирования на инцидент

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

(Например: Экзамен проводится по тест-билетам, каждый из которых содержит 10 вопросов и задачу. Каждый правильный ответ на вопрос в тесте оценивается 1 баллом, задача оценивается в 10 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.

2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 6 до 10 баллов

3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов.)

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы)	Код	Наименование
-------	-------------------------------	-----	--------------

	дисциплины	контролируемой компетенции	оценочного средства
1	Паттерны проектирования программного обеспечения	ОПК-13, ОПК-4.3	Тест, защита лабораторных работ, требования к курсовому проекту
2	Распределённое и параллельное программирование	ОПК-13, ОПК-4.3	Тест, защита лабораторных работ, требования к курсовому проекту
3	Высоконагруженные приложения	ОПК-13, ОПК-4.3	Тест, защита лабораторных работ, требования к курсовому проекту
4	Микросервисы	ОПК-13, ОПК-4.3	Тест, защита лабораторных работ, требования к курсовому проекту
5	Безопасный DevOps	ОПК-13, ОПК-4.3	Тест, защита лабораторных работ, требования к курсовому проекту
6	Паттерны проектирования программного обеспечения	ОПК-13, ОПК-4.3	Тест, защита лабораторных работ, требования к курсовому проекту

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методики выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Защита курсовой работы, курсового проекта или отчета по всем видам практик осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

Основная литература

Белугина, С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие / С. В. Белугина. — Санкт-Петербург : Лань, 2020. — 312 с. — ISBN 978-5-8114-4496-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/133920>

Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2022. — 235 с. — (Профессиональное образование). — ISBN 978-5-534-05047-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492496>

Дополнительная литература

Грувер, Г. Запуск и масштабирование DevOps на предприятии / Г. Грувер. — Москва : ДМК Пресс, 2018. — 80 с. — ISBN 978-5-97060-704-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/116130>

Скрынник, О. В. DevOps для ИТ-менеджеров: концентрированное структурированное изложение передовых идей / О. В. Скрынник. — 2-е изд. — Москва : ДМК Пресс, 2019. — 126 с. — ISBN 978-5-97060-692-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/112933>

Андреев, А. Е. Адаптивные технологии разработки программного обеспечения : учебное пособие / А. Е. Андреев, С. И. Кириносенко. — Волгоград : ВолгГТУ, 2015. — 96 с. — ISBN 978-5-9948-1979-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/157223>

Приемы объектно ориентированного проектирования. Паттерны проектирования : справочник / Э. Гамма, Р. Хелм, Р. Джонсон, Д. Влиссидес. — Москва : ДМК Пресс, 2007. — 368 с. — ISBN 5-93700-023-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/1220>

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

<http://att.nica.ru>

<http://www.edu.ru/>

<http://window.edu.ru/window/library>

<http://www.intuit.ru/catalog/>

<http://bibl.cchgeu.ru/MarcWeb2/ExtSearch.asp>
<https://cchgeu.ru/education/cafedras/kafsib/?docs>
<http://www.eios.vorstu.ru>
<http://e.lanbook.com/> (ЭБС Лань)
<http://IPRbookshop.ru/> (ЭБС IPRbooks)

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Специализированная лекционная аудитория, оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой.

Дисплейный класс, оснащенный компьютерными программами для проведения лабораторного практикума.

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Разработка прикладного программного обеспечения для компьютерных систем (связь, информационные и коммуникационные технологии)» читаются лекции, проводятся лабораторные работы, выполняется курсовой проект.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Методика выполнения курсового проекта изложена в учебно-методическом пособии. Выполнять этапы курсового проекта должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины производится проверкой курсового проекта, защитой курсового проекта.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополни-

	тельную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом, экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]