

Министерство науки и высшего образования Российской Федерации

ЧИТИНСКИЙ ИНСТИТУТ (ФИЛИАЛ)
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДЕН

на заседании кафедры информационных техно-
логий и высшей математики

24 февраля 2025 г. протокол № 6

Заведующий кафедрой

Л.И. Трухина



**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(ФОНД ОЦЕНОЧНЫХ СРЕДСТВ)
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**

Б1.Э.1 Разработка приложений для мобильных устройств

Направление подготовки: 38.03.05 Бизнес-информатика

Направленность (профиль): Цифровая экономика

Квалификация выпускника: бакалавр

Чита, 2025 г.

**Структура
фонда оценочных средств
по дисциплине «Разработка приложений для мобильных устройств»**

№ п/п	Этапы формирования компетенций	Перечень формируемых компетенций	ЗУНы (З.1, У1, Н1...)	Контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы	Описание показателей и критериев оценивания компетенций на различных этапах формирования, описания шкал оценивания
1	Мобильное программирование, платформы для разработки	ПК-4	З.Знать современные тенденции развития и применения инновационных цифровых компонент ИТ-инфраструктуры предприятия У.Уметь проектировать, разрабатывать и внедрять инновационные цифровые компоненты ИТ-инфраструктуры предприятия Н.Владеть навыками проектирования, разработки и внедрения инновационных цифровых компонент ИТ-инфраструктуры предприятия	Тест по основам проектирования и разработки МП	(процент правильных ответов * 20 баллов), округленных вверх до ближайшего целого числа, если процент правильных ответов меньше 50 – то 0 баллов (20)

2	<p>Программная платформа Xamarin. Основы XAML. Элементы управления. Концепция ресурсов в Xamarin Forms. Элементы компоновки. Типы страниц. Основы навигации. Стек навигации. Передача данных при навигации. Понятие привязки. Конвертеры значений. Жизненный цикл приложения. Манифест.</p>	ПК-4	<p>З.Знать современные тенденции развития и применения инновационных цифровых компонент ИТ-инфраструктуры предприятия У.Уметь проектировать, разрабатывать и внедрять инновационные цифровые компоненты ИТ-инфраструктуры предприятия Н.Владеть навыками проектирования, разработки и внедрения инновационных цифровых компонент ИТ-инфраструктуры предприятия</p>	<p>Лабораторная работа №1. Основы работы с UWP/Xamarin, язык разметки XAML. Основные элементы управления UWP/Xamarin. Графика</p>	<p>Полностью выполненная лабораторная работа - 18 баллов, частично - доля правильно выполненных заданий*18 баллов, если доля меньше 0,5 - 0 баллов (18)</p>
3	<p>Паттерн MVVM (Model - View - ViewModel). Интерфейс ICommand. Элементы управления данными. Основы LINQ. Файловая система. SQLite. Entity Framework</p>	ПК-4	<p>З.Знать современные тенденции развития и применения инновационных цифровых компонент ИТ-инфраструктуры предприятия У.Уметь проектировать, разрабатывать и внедрять инновационные цифровые компоненты ИТ-инфраструктуры предприятия Н.Владеть навыками проектирования, разра-</p>	<p>Лабораторная работа №2. Парадигма MVVM UWP/Xamarin</p>	<p>Полностью выполненная лабораторная работа - 20 баллов, частично - доля правильно выполненных заданий*20 баллов, если доля меньше 0,5 - 0 баллов (20)</p>

			ботки и внедрения инновационных цифровых компонент ИТ-инфраструктуры предприятия		
4	Создание веб-сервиса. Взаимодействие с веб-сервисом	ПК-4	3.Знать современные тенденции развития и применения инновационных цифровых компонент ИТ-инфраструктуры предприятия У.Уметь проектировать, разрабатывать и внедрять инновационные цифровые компоненты ИТ-инфраструктуры предприятия Н.Владеть навыками проектирования, разработки и внедрения инновационных цифровых компонент ИТ-инфраструктуры предприятия	Лабораторная работа №3. Работа с данными UWP/Xamarin	Полностью выполненная лабораторная работа - 20 баллов, частично - доля правильно выполненных заданий*20 баллов, если доля меньше 0,5 - 0 баллов (20)
5	Итого по текущей аттестации	ПК-4	3.Знать современные тенденции развития и применения инновационных цифровых компонент ИТ-инфраструктуры предприятия У.Уметь проектировать, разрабатывать и внедрять инновацион-	Разработка индивидуального мобильного приложения	Полностью выполненная лабораторная работа - 22 балла, частично - доля правильно выполненных заданий*22 балла, если доля меньше 0,5 - 0 баллов (22)

			ные цифровые компоненты ИТ-инфраструктуры предприятия Н. Владеть навыками проектирования, разработки и внедрения инновационных цифровых компонент ИТ-инфраструктуры предприятия		100
6	Промежуточная аттестация	ПК-4	З. Знать современные тенденции развития и применения инновационных цифровых компонент ИТ-инфраструктуры предприятия У. Уметь проектировать, разрабатывать и внедрять инновационные цифровые компоненты ИТ-инфраструктуры предприятия Н. Владеть навыками проектирования, разработки и внедрения инновационных цифровых компонент ИТ-инфраструктуры предприятия	Зачет	100

Министерство науки и высшего образования Российской Федерации
ЧИТИНСКИЙ ИНСТИТУТ (ФИЛИАЛ)
ФГБОУ ВО «БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Кафедра информационных технологий и высшей математики

Тест по основам проектирования и разработки мобильных приложений

Вопросы:

1. **Что такое мобильное приложение?**
 - a) Программа, работающая на настольном компьютере
 - b) Программа, разработанная для работы на мобильных устройствах
 - c) Веб-сайт, оптимизированный для мобильных устройств
 - d) Игра для консоли
2. **Какой из следующих языков программирования чаще всего используется для разработки приложений на Android?**
 - a) Swift
 - b) Java
 - c) C#
 - d) PHP
3. **Что такое UI (User Interface)?**
 - a) Логика приложения
 - b) Визуальные элементы, с которыми взаимодействует пользователь
 - c) База данных приложения
 - d) Серверная часть приложения
4. **Какой паттерн проектирования часто используется в мобильной разработке для разделения логики и представления?**
 - a) MVC (Model-View-Controller)
 - b) MVP (Model-View-Presenter)
 - c) MVVM (Model-View-ViewModel)
 - d) Все вышеперечисленные
5. **Какой из следующих инструментов используется для разработки приложений на iOS?**
 - a) Android Studio
 - b) Xcode
 - c) Visual Studio Code
 - d) Eclipse
6. **Что такое REST API?**
 - a) Протокол передачи данных между клиентом и сервером
 - b) Способ хранения данных в базе данных

- c) Язык программирования для создания веб-приложений
 - d) Метод шифрования данных
7. **Какой из следующих форматов данных чаще всего используется для обмена данными между клиентом и сервером в веб-приложениях?**
- a) XML
 - b) JSON
 - c) CSV
 - d) HTML
8. **Что такое "отзывчивый дизайн" (responsive design)?**
- a) Дизайн, который адаптируется к различным размерам экранов
 - b) Дизайн с фиксированными размерами элементов
 - c) Дизайн, который не требует тестирования на разных устройствах
 - d) Дизайн, который использует только текстовые элементы
9. **Какой из следующих фреймворков используется для кроссплатформенной разработки мобильных приложений?**
- a) React Native
 - b) AngularJS
 - c) Django
 - d) Ruby on Rails
10. **Что такое "прототипирование" в контексте проектирования приложений?**
- a) Создание окончательной версии приложения без тестирования
 - b) Создание предварительной модели приложения для тестирования идей и функций
 - c) Процесс написания кода приложения без дизайна интерфейса
 - d) Завершение проекта без обратной связи от пользователей
11. **Какой метод тестирования используется для проверки функциональности приложения?**
- a) Юзабилити-тестирование
 - b) Функциональное тестирование
 - c) Нагрузочное тестирование
 - d) Тестирование безопасности
12. **Что такое "пользовательский опыт" (UX)?**
- a) Визуальный стиль приложения
 - b) Общее впечатление пользователя от взаимодействия с приложением
 - c) Код, написанный для приложения
 - d) Процесс разработки интерфейса

- 13. Какой из следующих инструментов можно использовать для управления версиями кода?**
- a) Git
 - b) Docker
 - c) Jenkins
 - d) JIRA
- 14. Что такое "агильная методология" в разработке программного обеспечения?**
- a) Методология с фиксированным планом и сроками
 - b) Методология, основанная на гибком подходе к изменениям и итеративной разработке
 - c) Методология, которая не требует документации
 - d) Методология, используемая только в крупных проектах
- 15. Какой из следующих элементов является частью архитектуры MVC?**
- a) Модель
 - b) Представление
 - c) Контроллер
 - d) Все вышеперечисленные
- 16. Что такое "кэширование" в контексте мобильных приложений?**
- a) Процесс удаления ненужных данных
 - b) Хранение временных данных для ускорения доступа к ним
 - c) Процесс шифрования данных
 - d) Хранение всех данных в облаке
- 17. Какой из следующих методов позволяет пользователю взаимодействовать с элементами интерфейса?**
- a). События (Events)
 - b). Модели (Models)
 - c). Представления (Views)
 - d). Контроллеры (Controllers)
- 18. Что такое "интерфейс программирования приложений" (API)?**
- a). Набор правил и протоколов для взаимодействия между различными программами или компонентами.
 - b). Графический интерфейс пользователя.
 - c). База данных.
 - d). Язык программирования.
- 19. Какое из следующих утверждений о кроссплатформенной разработке верно?**
- a). Она позволяет создавать приложения только для одной платформы.

- b). Она требует написания отдельного кода для каждой платформы.
- c). Она позволяет использовать один код для нескольких платформ.
- d). Она не поддерживает доступ к нативным функциям устройства.

20. Что такое "доступность" в контексте мобильных приложений?

- a). Способность приложения работать только на новых устройствах.
- b). Уровень удобства использования приложения людьми с ограниченными возможностями.
- c). Количество пользователей, которые могут одновременно использовать приложение.
- d). Способность приложения загружаться быстро.

Ответы:

- 1. b
- 2. b
- 3. b
- 4. d
- 5. b
- 6. a
- 7. b
- 8. a
- 9. a
- 10. b
- 11. b
- 12. b
- 13. a
- 14. b
- 15. d
- 16. b
- 17. a
- 18. A
- 19. C
- 20. b

Лабораторная работа №1: Основы работы с UWP/Xamarin, XAML и графикой

Цели работы:

- 1. Ознакомиться с основами разработки приложений на UWP или Xamarin.

2. Научиться использовать язык разметки XAML для создания пользовательского интерфейса.
3. Изучить основные элементы управления и работу с графикой.

Задание:

1. **Создание проекта:**
 1. Создайте новый проект приложения на UWP или Xamarin.Forms в Visual Studio.
2. **Разработка пользовательского интерфейса:**
 1. Используя XAML, создайте основной экран приложения с следующими элементами управления:
 1. **Label**: для отображения текста.
 2. **TextBox**: для ввода данных пользователем.
 3. **Button**: для выполнения действия (например, отображение введенного текста).
 4. **ListView**: для отображения списка элементов (например, введенных пользователем строк).
 5. **Image**: для отображения графического элемента (например, изображения из ресурсов).
3. **Обработка событий:**
 1. Реализуйте обработчик события нажатия кнопки, который будет:
 1. Считывать текст из TextBox.
 2. Добавлять его в ListView.
 3. Отображать введенный текст в Label.
4. **Работа с графикой:**
 1. Добавьте возможность загрузки изображения из ресурсов приложения и его отображения в элементе Image.
 2. Реализуйте возможность изменения изображения при нажатии кнопки (например, переключение между двумя изображениями).
5. **Стилизация интерфейса:**
 1. Используйте стили и шаблоны для улучшения внешнего вида вашего приложения:
 1. Примените стили к элементам управления (например, измените цвет фона, шрифт и т.д.).
 2. Создайте шаблон для элемента списка в ListView.
6. **Тестирование:**
 1. Протестируйте приложение на эмуляторе или реальном устройстве.
 2. Убедитесь, что все элементы управления работают корректно и интерфейс выглядит привлекательно.
7. **Документация:**
 1. Подготовьте краткую документацию по вашему приложению, включающую:
 1. Описание функциональности.

2. Инструкции по запуску приложения.
3. Скриншоты интерфейса.

8. Презентация:

1. Подготовьте короткую презентацию о вашем приложении, в которой расскажете о его функциональности и процессе разработки.

Оценивание (18 баллов):

1. Полнота выполнения задания (30%)
2. Качество пользовательского интерфейса (30%)
3. Работоспособность приложения (20%)
4. Документация и презентация (20%)

Дополнительные рекомендации:

1. Используйте дополнительные элементы управления (например, Slider, ProgressBar) для улучшения функциональности вашего приложения.
2. Обратите внимание на адаптивный дизайн для обеспечения корректного отображения на разных устройствах.

Лабораторная работа №2: Парадигма MVVM в UWP/Xamarin

Цели работы:

1. Ознакомиться с архитектурой MVVM (Model-View-ViewModel).
2. Научиться разделять логику приложения и пользовательский интерфейс.
3. Реализовать простое приложение с использованием паттерна MVVM.

Задание:

1. **Создание проекта:**
 1. Создайте новый проект приложения на UWP или Xamarin.Forms в Visual Studio.
2. **Определение структуры проекта:**
 1. Создайте три папки в проекте:
 1. **Models:** для хранения классов модели.
 2. **ViewModels:** для хранения классов представления (ViewModel).
 3. **Views:** для хранения XAML-файлов представления.
3. **Создание модели (Model):**
 1. Создайте класс модели, который будет представлять данные вашего приложения. Например, это может быть класс Task, содержащий свойства:

1. Id (int)
2. Title (string)
3. IsCompleted (bool)
4. **Создание ViewModel:**
 1. Создайте класс ViewModel, который будет содержать коллекцию объектов модели и реализовывать логику взаимодействия с данными. Например, создайте класс TaskViewModel, который будет содержать:
 1. ObservableCollection<Task> для хранения списка задач.
 2. Команды для добавления и удаления задач.
 3. Свойства для привязки к элементам управления в интерфейсе.
5. **Создание представления (View):**
 1. Создайте XAML-файл для основного представления вашего приложения (например, MainPage.xaml).
 2. Используйте элементы управления, такие как ListView, TextBox, и Button, для отображения списка задач и взаимодействия с пользователем.
 3. Привяжите элементы управления к свойствам и командам из ViewModel.
6. **Реализация команд:**
 1. Реализуйте команды в ViewModel с использованием интерфейса ICommand. Например, создайте команды для добавления новой задачи и удаления выбранной задачи из списка.
7. **Двусторонняя привязка данных:**
 1. Убедитесь, что данные обновляются автоматически при изменении свойств в ViewModel благодаря механизму уведомлений об изменениях (INotifyPropertyChanged).
8. **Тестирование:**
 1. Протестируйте приложение на эмуляторе или реальном устройстве.
 2. Убедитесь, что все функции работают корректно и данные обновляются в интерфейсе при изменении в ViewModel.
9. **Документация:**
 1. Подготовьте краткую документацию по вашему приложению, включающую:
 1. Описание функциональности.
 2. Инструкции по запуску приложения.
 3. Скриншоты интерфейса.
10. **Презентация:**
 1. Подготовьте короткую презентацию о вашем приложении, в которой расскажете о его функциональности и процессе разработки с акцентом на использование паттерна MVVM.

Оценивание (20 баллов):

1. Полнота выполнения задания (30%)
2. Качество реализации паттерна MVVM (30%)
3. Работоспособность приложения (20%)
4. Документация и презентация (20%)

Дополнительные рекомендации:

1. Используйте дополнительные элементы управления и стилизацию для улучшения пользовательского интерфейса.
2. Рассмотрите возможность добавления функциональности сохранения данных локально (например, с использованием SQLite).

Лабораторная работа №3. Работа с данными в UWP/Xamarin

Цели работы:

1. Ознакомиться с различными способами хранения и обработки данных в приложениях UWP/Xamarin.
2. Научиться работать с локальным хранилищем (например, SQLite).
3. Изучить основы работы с RESTful API для получения данных.

Задание:

1. **Создание проекта:**
 1. Создайте новый проект приложения на UWP или Xamarin.Forms в Visual Studio.
2. **Работа с локальным хранилищем (SQLite):**
 1. Добавьте пакет NuGet SQLite-net-pcl или Microsoft.Data.Sqlite для работы с SQLite.
 2. Создайте класс модели, например, Note, который будет содержать свойства:
 1. Id (int)
 2. Title (string)
 3. Content (string)
 4. CreatedDate (DateTime)
 3. Реализуйте класс для управления базой данных, который будет включать методы для добавления, удаления и получения записей.
3. **Создание пользовательского интерфейса:**
 1. Создайте XAML-страницу для отображения списка записей и формы для добавления новой записи.
 2. Используйте элементы управления, такие как ListView, Entry, и Button.
4. **Реализация CRUD операций:**
 1. Реализуйте функциональность добавления новой записи через форму.
 2. Отображайте список записей в ListView.

3. Реализуйте возможность удаления заметки из списка.
5. **Работа с RESTful API:**
 1. Выберите открытое API (например, JSONPlaceholder) и создайте класс для работы с API.
 2. Реализуйте метод для получения данных из API (например, список постов или пользователей).
 3. Отобразите полученные данные на отдельной странице вашего приложения.
6. **Асинхронные операции:**
 1. Используйте асинхронные методы (`async/await`) при работе с базой данных и API, чтобы не блокировать пользовательский интерфейс.
7. **Тестирование:**
 1. Протестируйте приложение на эмуляторе или реальном устройстве.
 2. Убедитесь, что все функции работают корректно и данные отображаются правильно.
8. **Документация:**
 1. Подготовьте краткую документацию по вашему приложению, включающую:
 1. Описание функциональности.
 2. Инструкции по запуску приложения.
 3. Скриншоты интерфейса.
9. **Презентация:**
 1. Подготовьте короткую презентацию о вашем приложении, в которой расскажете о его функциональности и процессе разработки.

Оценивание (20 баллов):

1. Полнота выполнения задания (30%)
2. Качество реализации работы с данными (30%)
3. Работоспособность приложения (20%)
4. Документация и презентация (20%)

Дополнительные рекомендации:

1. Рассмотрите возможность добавления функциональности редактирования заметок.
2. Используйте MVVM-паттерн для разделения логики приложения и пользовательского интерфейса.
3. Обратите внимание на обработку ошибок при работе с API и базой данных.

Лабораторная работа №4. Разработка индивидуального мобильного приложения

Цели работы:

1. Применить полученные знания по разработке мобильных приложений на практике.
2. Разработать функциональное и пользовательски-дружественное приложение.
3. Ознакомиться с процессом проектирования, разработки и тестирования мобильного приложения.

Задание:

1. Выбор темы приложения:

1. Выберите тему для вашего мобильного приложения. Это может быть:
 1. Приложение для заметок или задач.
 2. Приложение для отслеживания привычек или фитнеса.
 3. Приложение для изучения языков.
 4. Игровое приложение (простая игра).
 5. Приложение для управления финансами.
2. Обоснуйте выбор темы и целевую аудиторию вашего приложения.

2. Проектирование интерфейса:

1. Создайте прототип интерфейса вашего приложения с помощью инструментов, таких как Figma, Adobe XD или Sketch.
2. Определите основные экраны и навигацию между ними (например, экран входа, главный экран, экран настроек и т.д.).
3. Подготовьте описание пользовательского опыта (UX) и пользовательского интерфейса (UI).

3. Разработка приложения:

1. Создайте новый проект в UWP или Xamarin.Forms.
2. Реализуйте основные функции вашего приложения согласно прототипу:
 1. Используйте MVVM-паттерн для организации кода (если это применимо).
 2. Реализуйте локальное хранилище данных (например, SQLite) или интеграцию с API (если это необходимо).
 3. Обеспечьте возможность взаимодействия пользователя с приложением через элементы управления (кнопки, списки, формы ввода и т.д.).

4. Тестирование:

1. Проведите тестирование вашего приложения на эмуляторе или реальном устройстве.

2. Убедитесь, что все функции работают корректно и интерфейс удобен для пользователя.
5. **Документация:**
 1. Подготовьте документацию по вашему приложению, включающую:
 1. Описание функциональности.
 2. Инструкции по установке и запуску приложения.
 3. Скриншоты интерфейса и описание основных функций.
6. **Презентация:**
 1. Подготовьте короткую презентацию о вашем приложении (5-10 минут), в которой расскажете о его функциональности, процессе разработки и полученных результатах.
 2. Включите демонстрацию работы приложения.

Оценивание (22 балла):

1. Полнота выполнения задания (30%)
2. Качество реализации функциональности (30%)
3. Пользовательский интерфейс и опыт (20%)
4. Документация и презентация (20%)

Дополнительные рекомендации:

1. Рассмотрите возможность добавления дополнительных функций или улучшений на основе обратной связи от пользователей.
2. Обратите внимание на производительность и оптимизацию вашего приложения.
3. Используйте современные библиотеки и инструменты для улучшения функциональности вашего приложения.

Вопросы к зачету

1. SQLite. Применение в UWP. Возможности. Entity Framework
2. Адаптивный дизайн и код. Представления XAML
3. Архитектура операционной системы Android
4. Архитектура приложений UWP
5. Архитектура приложений Xamarin
6. Виды приложений Android и их структура
7. Двухмерная графика. Базовые графические примитивы.
8. Диаграммы UML.
9. Установка разработанного программного обеспечения (манифест, магазины)
10. Основные понятия и структура проекта информационной системы (ИС)
11. Основные принципы дизайна интерфейса мобильных/универсальных приложений
12. Основные типы разработки мобильных приложений.

13. Основные элементы XAML и их атрибуты
14. Основные языки и парадигмы программирования для разработки мобильных приложений
15. Понятие привязки, способы привязки данных
16. Работа с аудио и видео. Фоновые задачи.
17. Работа с графикой и мультимедиа в UWP. Основные приемы и используемые классы
18. Работа с файловой системой на примере UWP или Android. Основные концепции, классы и объекты, принципы
19. Разграничение прав пользователей ИС
20. Стили и шаблоны UWP. Способы создания, примеры использования
21. Тестирование и отладка ИС
22. Элементы управления данными. ObservableCollection
23. Элементы управления данными. Их поведение и предназначение, примеры использования
24. Этапы жизненного цикла программного обеспечения. Модели жизненного цикла

Образец билета на зачет

<p>Читинский институт (филиал) ФГБОУ ВО Байкальский государственный университет Кафедра Информационных технологий и высшей математики</p>	<p>БИЛЕТ НА ЗАЧЁТ №1 по дисциплине Разработка приложений для мобильных устройств Направление подготовки Бизнес-информатика</p>
<p>1. Тест (40 баллов).</p> <p>2. Создайте UWP/Xamarin/Android приложение и продемонстрируйте работу элементов (назначение, возможности, свойства, кастомизацию и пр.) в соотв. с вариантом (30 баллов).</p> <p>3. Разработайте мобильное приложение в соотв. с вариантом (30 баллов).</p>	
<p>Билет составила Богатикова Е.О. Утверждаю /зав. кафедрой Трухина Л.И./</p>	

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Система критериев оценки определяет оценку успеваемости по каждому заданию (вопросу) экзаменационного билета или заданию для зачета с использованием интервальной шкалы баллов, применяемой в привязке к рейтинговой 100-балльной системе.

ОЦЕНКА ОТВЕТА НА ТЕОРЕТИЧЕСКИЙ ВОПРОС В УСТНОЙ ИЛИ ПИСЬМЕННОЙ ФОРМЕ:

Оценка «отлично» / «зачтено» (91-100 баллов) выставляется при соблюдении следующих условий: Ответ отличается глубиной и полнотой, свободным владением понятийно-категориальным (терминологическим) аппаратом изученной дисциплины. Отражает знание не только основной, но и дополнительной литературы. Приведены примеры, отражающие умение связать теорию с практикой. Ответ изложен логически последовательно, грамотно и корректно.

Оценка «хорошо» / «зачтено» (76-90 баллов) выставляется при соблюдении следующих условий: Ответ отличается полнотой, владением понятийно-категориальным (терминологическим) аппаратом изученной дисциплины, но в ответе могут присутствовать неточности. Отражает знание основной литературы. Приведены примеры, отражающие умение связать теорию с практикой. Ответ изложен логически последовательно, грамотно и корректно, но недостаточно аргументирован.

Оценка «удовлетворительно» / «зачтено» (61-75 баллов) выставляется при соблюдении следующих условий: В ответе отражено знание понятийно-категориального (терминологического) аппарата изучаемой дисциплины, но присутствуют отдельные ошибки и неточности. Ответ характеризуется недостаточным знанием рекомендованной литературы. Примеры, отражающие умение связать теорию с практикой, тривиальны, либо отсутствуют. Ответ неполный, носит фрагментарный, непоследовательный характер.

Оценка «неудовлетворительно» / «не зачтено» (0-60 баллов) выставляется при соблюдении следующих условий: Ответ характеризуется незнанием, либо фрагментарным представлением о понятийно-категориальном аппарате дисциплины, содержит множество ошибок. Примеры и иллюстрации отсутствуют. Ответ логически непоследователен.

ОЦЕНКА ВЫПОЛНЕНИЯ ЗАДАНИЯ В ФОРМЕ CASE-STUDY (СИТУАЦИИ)

Оценка «отлично» / «зачтено» (91-100 баллов) выставляется при соблюдении следующих условий: Четкая формулировка проблемы. Полное и соответствующее ситуации решение, основанное на знании правовых норм и технологий (опыте), применяемых в реальных организациях (известных компаниях). Предполагаемые действия описаны логично и последовательно. Даны дополнительные авторские комментарии и предложения к решению ситуации.

Оценка «хорошо» / «зачтено» (76-90 баллов) выставляется при соблюдении следующих условий: Понимание сути проблемы, но ее формулирование затруднено. Решение соответствует ситуации, отражает знание правовых норм и опыт работы других организаций при решении подобных ситуаций. Логика и последовательность действий не нарушены.

Оценка «удовлетворительно» / «зачтено» (61-75 баллов) выставляется при соблюдении следующих условий: Проблема не сформулирована. Приведен набор действий, потенциально способствующих улучшению ситуации и решению проблемы.

Оценка «неудовлетворительно» / «не зачтено» (0-60 баллов) выставляется при соблюдении следующих условий: Предложенный перечень мероприятий не соответствует

ситуации.

ОЦЕНКА РЕШЕНИЯ ЗАДАЧИ

Оценка «отлично» / «зачтено» (91-100 баллов) выставляется при соблюдении следующих условий: Полное верное решение - оценивается в n баллов (n – максимальное количество баллов за решение задачи в структуре экзаменационного билета/задания).

Оценка «хорошо» / «зачтено» (76-90 баллов) выставляется при соблюдении следующих условий: Верное решение; имеются небольшие недочеты, в целом не влияющие на решение – оценивается в диапазоне от $0,76*n$ баллов до $0,9*n$ баллов (n – максимальное количество баллов за решение задачи в структуре экзаменационного билета/задания).

Оценка «удовлетворительно» / «зачтено» (61-75 баллов) выставляется при соблюдении следующих условий: Решение в целом верное; однако оно содержит ряд ошибок, либо не учитывает отдельных случаев, но может стать правильным после некоторых исправлений или дополнений – оценивается в диапазоне от $0,61*n$ баллов до $0,75*n$ баллов (n – максимальное количество баллов за решение задачи в структуре экзаменационного билета/задания).

Оценка «неудовлетворительно» / «не зачтено» (0-60 баллов) выставляется при соблюдении следующих условий: Решение неверное; изначально выбран неверный ход решения, или решение отсутствует – оценивается в 0 баллов.

ОЦЕНКА ВЫПОЛНЕНИЯ ТЕСТОВОГО ЗАДАНИЯ

Подсчитывается доля набранных баллов в максимальной сумме баллов за все задания теста:

– Каждый правильный ответ на тестовый вопрос (тип выборочный, одинарный, множественный, открытый) оценивается в m баллов (число m определяется путем деления максимального количества баллов за выполнение теста в структуре экзаменационного билета/задания на количество тестовых заданий);

– Каждый частично правильный ответ на тестовый вопрос (тип выборочный, множественный, открытый) оценивается в $m/2$ баллов независимо от соотношения правильно/неправильно выбранных вариантов (число m определяется путем деления максимального количества баллов за выполнение теста в структуре экзаменационного билета/задания на количество тестовых заданий);

– Каждый неправильный ответ на тестовый вопрос (тип выборочный, одинарный) оценивается в 0 баллов.

Оценка «отлично»/ «зачтено» (91-100 баллов) выставляется, если доля набранных баллов составляет 91-100%.

Оценка «хорошо»/ «зачтено» (76-90 баллов), если доля набранных баллов составляет 76-90%.

Оценка «удовлетворительно»/ «зачтено» (61-75 баллов), если доля набранных баллов составляет 61-75%.

Оценка «неудовлетворительно»/ «не зачтено» (0-60 баллов), если доля набранных баллов составляет не более 60%.