



Федеральное государственное бюджетное образовательное учреждение высшего образования  
«ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ ДВАЖДЫ ГЕРОЯ  
СОВЕТСКОГО СОЮЗА, ЛЕТЧИКА-КОСМОНАВТА А.А. ЛЕОНОВА»

**УТВЕРЖДАЮ**  
**И. о. проректора**  
**А.В. Троицкий**

«\_\_» \_\_\_\_\_ 2023 г.

**ИНСТИТУТ  
ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**РАБОЧАЯ ПРОГРАММА  
ДИСЦИПЛИНЫ  
«РАЗРАБОТКА КРОСС-ПЛАТФОРМЕННЫХ ПРИЛОЖЕНИЙ»**

**Направление подготовки:** 09.03.04 Программная инженерия

**Направленность (профиль):** Проектирование и разработка программного обеспечения

**Уровень высшего образования:** бакалавриат

**Форма обучения:** очная

Королев  
2023

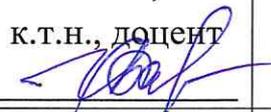
Рабочая программа является составной частью основной профессиональной образовательной программы и проходит рецензирование со стороны работодателей в составе основной профессиональной образовательной программы. Рабочая программа актуализируется и корректируется ежегодно.

**Автор: к.т.н., с.н.с. Парафейников И.В. Рабочая программа дисциплины: Разработка кросс-платформенных приложений – Королев МО: «Технологический университет», 2023 г.**

Рецензент: д.т.н., профессор Артюшенко В.М.

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки бакалавров 09.03.04 Программная инженерия, учебного плана, утвержденного Ученым советом Технологического университета, Протокол № 9 от 11 апреля 2023 года..

**Рабочая программа рассмотрена и одобрена на заседании кафедры:**

Заведующий кафедрой (ФИО, ученая степень, звание, подпись)	Баранова О.М., к.т.н., доцент 			
Год утверждения (переутверждения)	2023	2024	2025	2026
Номер и дата протокола заседания кафедры	№ 7 от 03.04.2023			

**Рабочая программа согласована:**

Руководитель ОПОП  О.М. Баранова, к.т.н.

**Рабочая программа рекомендована на заседании УМС:**

Год утверждения (переутверждения)	2023	2024	2025	2026
Номер и дата протокола заседания УМС	№ 5 от 11.04.2023 г.			

## **1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОПОП**

Целью дисциплины "Разработка кросс-платформенных приложений" является изучение основ разработки кроссплатформенных приложений, включающие в себя:

- Особенности разработки кроссплатформенных приложений;
- Принципы разработки кроссплатформенных приложений;
- Архитектуру и жизненный цикл кроссплатформенных приложений;
- Работу основных компонентов кроссплатформенных приложений;
- Технологии кроссплатформенной разработки;
- Различные библиотеки и фреймворки, применяемые при кросс-платформенной разработке.

Дисциплина направлена на формирование следующих компетенций:

- (ПК-1) Способен определять и анализировать требования к программному обеспечению и возможность их реализации, в том числе разрабатывать технические спецификации на программное обеспечение и его компоненты;
- (ПК-3) Способен управлять архитектурой программного обеспечения, в том числе интегрированного;
- (ПК-4) Способен разрабатывать программные модули и компоненты программных продуктов;
- (ПК-5) Способен выполнять работы по созданию, модификации и сопровождению интеграционных решений;
- (ПК-6) Способен управлять работами по созданию и модификации информационных ресурсов;
- (ПК-7) Способен анализировать качество программного обеспечения и определять методы его повышения.

Задача дисциплины - получить обучающимся необходимый набор знаний и навыков для дальнейшей работы в области кроссплатформенной разработки и более глубокого изучения данной отрасли программирования.

Показатель освоения компетенции отражают следующие индикаторы:

### **Трудовые действия:**

- Определяет требования к программному обеспечению и возможность их реализации;
- Разрабатывает технические спецификации на программное обеспечение и его компоненты;
- Выявляет и согласовывает требования к программной системе с точки зрения архитектуры;
- Выбирает и моделирует архитектурное решение для реализации программной системы;

- Разрабатывает программные модули и компоненты программных продуктов;
- Осуществляет поддержку процесса согласования требований к интеграционному решению;
- Конфигурирует интеграционное решение;
- Анализирует требования к информационным ресурсам, разрабатывает технические спецификации на информационные ресурсы;
- Проектирует, разрабатывает и тестирует информационные ресурсы;
- Производит анализ качества программного обеспечения;
- Определяет и применяет методы повышения качества программного обеспечения;

**Необходимые умения:**

- Анализирует требования к программному обеспечению;
- Определяет состав информации для разработки технических спецификаций на программное обеспечение и его компоненты;
- Определяет требования к программному продукту с точки зрения архитектуры;
- Применяет различные способы и методы моделирования архитектурного решения;
- Применяет современные технологии, методы и средства разработки программных модулей и компонентов программных продуктов;
- Знает основные принципы, способы и методы проектирования программного обеспечения;
- Знает состав проектной документации и основные принципы ее разработки;
- Применяет принципы определения требований к интеграционному решению;
- Использует основные методы, способы и средства интеграции программных продуктов;
- Применяет основные принципы разработки технических спецификаций на информационные ресурсы;
- Разрабатывает модели информационных ресурсов;
- Проводит тестирование информационных ресурсов;
- Использует различные методы и способы анализа качества и повышения его уровня.

**Необходимые знания:**

- Знает принципы определения требований к программному обеспечению;
- Знает принципы и правила разработки технических спецификаций на программное обеспечение и его компоненты;
- Знает понятие архитектуры программного обеспечения;
- Знает основные особенности архитектуры интегрированного программного обеспечения;

- Знает современные технологии, методы и средства разработки программных модулей и компонентов программных продуктов;
- Знает особенности разработки различных программных модулей и компонентов программных продуктов
- Знает принципы определения требований к интеграционному решению;
- Знает основные методы, способы и средства интеграции программных продуктов
- Знает основные принципы разработки технических спецификаций на информационные ресурсы;
- Знает основные принципы тестирования информационных ресурсов;
- Знает концепции и атрибуты качества;
- Знает методы и способы анализа качества и повышения его уровня.

## **2. Место дисциплины в структуре ОПОП ВО**

Дисциплина «Разработка кросс-платформенных приложений» относится к дисциплинам по выбору части, формируемой участниками образовательных отношений, основной профессиональной образовательной программы подготовки по направлению подготовки бакалавриата 09.03.04 «Программная инженерия».

Дисциплина реализуется кафедрой прикладного программного обеспечения.

Изучение данной дисциплины базируется на знаниях и умениях, полученных в рамках ранее изученных дисциплин «Основы системотехники», «Базы данных», «Операционные системы, среды и оболочки», «Языки программирования высокого уровня», «Технологии и среды программирования», «Объектно-ориентированное программирование», «Параллельные вычисления», «Проектирование программного обеспечения», «Теория систем и системный анализ», «Теория случайных процессов» и компетенциях УК-1, УК-2, ОПК-2, ОПК-3, ОПК-6, ОПК-7, ОПК-8, ПК-1, ПК-2, ПК-5, ПК-6, ПК-7.

## **2. Объем дисциплины и виды учебной работы**

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

**Таблица 1**

<b>Виды занятий</b>	<b>Всего часов</b>	<b>Семестр 5</b>
<b>Общая трудоемкость</b>	<b>180</b>	<b>180</b>
<b>Аудиторные занятия</b>	<b>80</b>	<b>80</b>
Лекции (Л)	<b>32</b>	<b>32</b>
Практические занятия (ПЗ)	<b>48</b>	<b>48</b>

Лабораторные работы (ЛР)	–	–
<b>Практическая подготовка</b>	<b>32</b>	<b>32</b>
<b>Самостоятельная работа</b>	<b>100</b>	<b>100</b>
<b>Курсовые работы (проекты)</b>	<b>+</b>	<b>+</b>
<b>Расчетно-графические работы</b>	<b>–</b>	<b>–</b>
<b>Контрольная работа, домашнее задание</b>	<b>+</b> <b>–</b>	<b>+</b> <b>–</b>
<b>Текущий контроль знаний (7 - 8, 14 - 15 недели)</b>	<b>Тест</b>	<b>Тест</b>
<b>Вид итогового контроля</b>	<b>Экзамен</b>	<b>Экзамен</b>

## 4. Содержание дисциплины

### 4.1. Темы дисциплины и виды занятий

Таблица 2

Наименование тем	Лекции, час	Практические занятия, Час	Занятия в интерактивной форме, час	Код компетенций
Тема 1. Основные понятия. Нативный и кросс-платформенный подходы к разработке программного обеспечения.	2	4	2	ПК-1, ПК-3, ПК-4, ПК-5, ПК-6, ПК-7
Тема 2. Инструменты кроссплатформенного программного обеспечения	8	12	4	
Тема 3. Архитектура платформы .NET Core. Разработка графических приложений с использованием .NET Core.	8	12	4	
Тема 4. Введение в разработку с использованием Xamarin. Платформно-зависимая программная логика в Xamarin.	8	12	4	
Тема 5. Паттерны разработки мобильного программного обеспечения.	6	8	2	
<b>Итого:</b>	<b>32</b>	<b>48</b>	<b>16</b>	

### 4.2. Содержание тем дисциплины

**Тема 1. Основные понятия. Нативный и кросс-платформенный подходы к разработке программного обеспечения.**

Цели и задачи дисциплины, история. Понятие кросс-платформенности, различные виды кросс-платформенности. История развития кросс-платформенного программирования. Определение кросс-платформенности. Кросс-платформенность на уровне компилятора. Интерпретируемые языки

программирования и кросс-платформенность. Виртуальные машины, сравнение наиболее распространенных решений. Нативный и кросс-платформенный подходы к разработке ПО. Преимущества и недостатки Сравнительный анализ нативного и кросс-платформенного подхода к разработке. Архитектурные особенности кросс-платформенного ПО. Сложности кросс-платформенной разработки. Затраты на нативную разработку ПО.

## **Тема 2. Инструменты кросс-платформенного программного обеспечения**

Инструменты КППО. Различные варианты классификации инструментов, сравнительный анализ. Кросс-платформенная разработка консольных приложений, приложений с графическим пользовательским интерфейсом, интерактивных визуализаций. Кросс-платформенность для мобильных устройств и для персональных компьютеров. Стандартизированные библиотеки среды выполнения. Сравнение, преимущества и недостатки.

## **Тема 3. Архитектура платформы .NET Core. Разработка графических приложений с использованием .NET Core.**

Модульная платформа .NET Core. Обзор архитектуры. Анализ среды выполнения Различия между .NET Standard и .NET Core. Модульность и кросс-платформенность в .NET Core. Технология ASP.NET Core. Обзор .NET Core CLR. Runtime, JIT-компилятор, CoreLib. Сценарии развертывания (FDD, SCD), .NET Core Runtime. Верстка адаптивного графического пользовательского интерфейса. Язык XAML Краткий обзор языков разметки пользовательских интерфейсов. Проблема адаптивности пользовательского интерфейса. Элементы и атрибуты, пространства имен, компоновка, привязка, стили и шаблоны. Реализация платформу-зависимой функциональности.

## **Тема 4. Введение в разработку с использованием Xamarin. Платформу-зависимая программная логика в Xamarin.**

Платформа кросс-платформенной мобильной разработки Xamarin. Обзор архитектуры Xamarin.Android, Xamarin.IOS, компиляторы, расширения для различных сред разработки. Оберточная архитектура Xamarin. Слои приложения Xamarin. Xamarin, платформу-зависимые и платформу-независимые компоненты. Методы расширения, платформу-зависимые классы, объявления xmlns.

## **Тема 5. Паттерны разработки мобильного программного обеспечения.**

Классические паттерны проектирования в контексте Xamarin. Управление зависимостями для конкретных платформ. Инверсия зависимостей. Паттерны проектирования "фабрика" и Model-View-View Model (MVVM). Взаимодействие Objective-C, Java, C и C++. Надежная библиотека базовых классов (BCL). Поддержка кроссплатформенных мобильных приложений.

## **5. Перечень учебно-методического обеспечения для самостоятельной работы по дисциплине**

«Методические указания для обучающихся по освоению дисциплины»

## **6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине**

Структура фонда оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведена в Приложении 1 к настоящей Рабочей программе.

## **7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

### **Основная литература:**

1. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения. Часть 1: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова; Южный федеральный университет. - Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2019. - 139 с. - ISBN 978-5-9275-3367-1. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1088203>. - Режим доступа: по подписке.

2. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения. Часть 2: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова; Южный федеральный университет. - Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2019. - 168 с. - ISBN 978-5-9275-3368-8. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1088205>. - Режим доступа: по подписке.

3. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения: в 3 ч. Часть 3: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова; Южный федеральный университет. - Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2021. - 214 с. - ISBN 78-5-9275-3628-3. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1894418>. - Режим доступа: по подписке.

4. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке C++: учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев ; под ред. Л.Г. Гагариной. - Москва: ФОРУМ: ИНФРА-М, 2021. - 512 с. + Доп. материалы [Электронный ресурс]. - (Среднее профессиональное образование). - ISBN 978-5-8199-0699-6. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1172261>. - Режим доступа: по подписке.

## **Дополнительная литература:**

1. Кузин, А. В. Программирование на языке Си: учебное пособие / А.В. Кузин, Е.В. Чумакова. - Москва: ФОРУМ: ИНФРА-М, 2023. - 143 с. - (Среднее профессиональное образование). - ISBN 978-5-00091-556-1. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1878382>. - Режим доступа: по подписке.
2. Кузин, А. В. Программирование на языке Си: учебное пособие / А.В. Кузин, Е.В. Чумакова. - Москва: ФОРУМ: ИНФРА-М, 2021. - 144 с. - (Высшее образование). - ISBN 978-5-00091-066-5. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1222078>. - Режим доступа: по подписке.
3. Бедердинова, О. И. Программирование на языках высокого уровня: учеб. пособие / О.И. Бедердинова, Т.А. Минеева, Ю.А. Водовозова. - Москва: ИНФРА-М, 2019. - 159 с. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1044396>. - Режим доступа: по подписке.

## **Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

### **Интернет-ресурсы:**

1. <http://www.ru.wikipedia.org>
2. <http://intuit.ru>
3. <http://metanit.com>
4. <http://developer.android.com>
5. <https://biblioclub.ru/> - Университетская библиотека ONLINE.
6. <https://www.java.com/ru/> - Сайт разработчика Java
7. <https://www.rucont.ru/> - Электронная библиотека Руконт.
8. <https://znanium.com/> - Электронная библиотека издательства Инфра-М.
9. <https://e.lanbook.com/> - Электронная библиотека Лань.
10. <https://www.jetbrains.com/ru-ru/idea/> - Сайт разработчика IntelliJ IDEA.
11. <https://github.com/> - Веб-сервис для хостинга IT-проектов и их совместной разработки.
12. <https://www.eclipse.org/> - Сайт разработчика Eclipse
13. <https://jug.ru/> - Сайт Java-разработчиков

## **8. Методические указания для обучающихся по освоению дисциплины**

Методические указания для обучающихся по освоению дисциплины приведены в Приложении 2 к настоящему Положению.

## **9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

### **Перечень программного обеспечения:**

Msoffice, Double Commander, VS Code, Eclipse IDE, IntelliJ IDEA, JDK Java SE8 и выше, Notepad++, Android Studio, Open Server, Mozilla Firefox, Google Chrome, Adobe Reader XI или Adobe Acrobat Reader DC, Kaspersky Endpoint Security для Windows.

**Информационные справочные системы:**

1. Электронные ресурсы образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Разработка кросс-платформенных приложений».

**10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**

**Лекционные занятия:**

- аудитория, оснащенная презентационной техникой (проектор, экран);
- комплект электронных презентаций/слайдов.

**Практические занятия:**

- учебный класс, оснащенный вычислительной техникой (ПК) и доступом к Интернет-ресурсам.
- рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
- рабочие места студентов, оснащенные компьютерами с доступом в Интернет.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И  
ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ  
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО  
ДИСЦИПЛИНЕ  
«РАЗРАБОТКА КРОСС-ПЛАТФОРМЕННЫХ ПРИЛОЖЕНИЙ»  
(Приложение 1 к рабочей программе)**

**Направление подготовки: 09.03.04 Программная инженерия**

**Профиль: проектирование и разработка программного обеспечения**

**Уровень высшего образования: бакалавриат**

**Форма обучения: очная**

## 1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п/п	Индекс компетенции	Содержание компетенции (или ее части)*	Раздел дисциплины, обеспечивающий формирование компетенции (или ее части)	В результате изучения раздела дисциплины, обеспечивающего формирование компетенции (или ее части), обучающийся должен:		
				Необходимые знания	Необходимые умения	Трудовые действия
1	ПК-1 ПК-3	Способен определять и анализировать требования к программному обеспечению и возможность их реализации, в том числе разрабатывать технические спецификации на программное обеспечение и его компоненты. Способен управлять архитектурой программного обеспечения, в том числе интегрированного	<b>Тема 1.</b> Основные понятия. Нативный и кросс-платформенный подходы к разработке программного обеспечения. <b>Тема 2.</b> Инструменты кросс-платформенного программного обеспечения	Знает принципы определения требований к программному обеспечению. Знает понятие архитектуры программного обеспечения.	Анализирует требования к программному обеспечению. Применяет различные способы и методы моделирования архитектурного решения.	Разрабатывает технические спецификации на программное обеспечение и его компоненты. Выбирает и моделирует архитектурное решение для реализации программной системы.
2	ПК-4 ПК-5	Способен разрабатывать программные модули и компоненты программных продуктов. Способен выполнять работы по созданию, модификации и сопровождению интеграционных решений.	<b>Тема 3.</b> Архитектура платформы .NET Core. Разработка графических приложений с использованием .NET Core. <b>Тема 4.</b> Введение в разработку с использованием Xamarin. Платформозависимая программная логика в Xamarin.	Знает современные технологии, методы и средства разработки программных модулей и компонентов программных продуктов. Знает основные методы, способы и средства интеграции программных продуктов.	Применяет современные технологии, методы и средства разработки программных модулей и компонентов программных продуктов. Использует основные методы, способы и средства интеграции программных продуктов.	Разрабатывает программные модули и компоненты программных продуктов. Осуществляет поддержку процесса согласования требований к интеграционному решению; Конфигурирует интеграционное решение.
3.	ПК-6 ПК-7	Способен управлять работами по созданию и модификации информационных ресурсов; Способен анализировать	<b>Тема 5.</b> Паттерны разработки мобильного программного обеспечения.	Знает основные принципы тестирования информационных ресурсов. Знает концепции и атрибуты	Разрабатывает модели информационных ресурсов. Использует различные методы и способы	Проектирует, разрабатывает и тестирует информационные ресурсы. Производит анализ

	качество программного обеспечения и определять методы его повышения		качества.	анализа качества и повышения его уровня.	качества программного обеспечения.
--	---	--	-----------	--	------------------------------------

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

Код компетенции	Инструмент, оценивающий сформированность компетенции	Показатель оценивания компетенции	Критерии оценки
ПК-1, ПК-3, ПК-4, ПК-5, ПК-6, ПК-7	Практическое задание	<p>А) полностью сформирована (компетенция освоена на <u>высоком</u> уровне) – 5 баллов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> <li>• компетенция освоена на <u>продвинутом</u> уровне – 4 балла;</li> <li>• компетенция освоена на <u>базовом</u> уровне – 3 балла;</li> </ul> <p>В) не сформирована (компетенция не сформирована) – 2 и менее баллов</p>	<p>1. Проводится в форме практического задания</p> <p>2. Время, отведенное на процедуру – 20-30 мин.</p> <p>Неявка – 0.</p> <p>Критерии оценки:</p> <p>1. Соответствие ответа заявленной тематике (0-5 баллов).</p> <p>Максимальная сумма баллов - 5 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля.</p> <p>Оценка проставляется в электронный журнал.</p>

## 3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

### 3.1. Примерная тематика практических заданий:

**Задание 1.** Напишите консольную математическую игру. В этой игре в каждом раунде игрока просят вычислить не сложный математический пример (например,

2+2, 3-4 и т.п.). Числовые значения и знак операции должны генерироваться случайным образом. За каждый правильно вычисленный пример следует начислять баллы, а за ошибки отнимать количество попыток продолжения игры.

**Задание 2.** Самостоятельно напишите консольную интерактивную игру.

Некоторые требования к разработанной вами игре:

- наличие начала игры, например после запуска в консоли крупно выводится название игры и выдается приветствие
- возможность задания имени игрока
- наличие у игрока и у его противника таких характеристик, как: очки жизни, удача, сила удара, оружие и т.п.
- в каждом раунде игры характеристики противника должны генерироваться случайным образом
- в начале каждого раунда или перед ходом игрока ему должна быть предоставлена возможность убежать, которая определяется случайным образом
- за каждую победу игроку начисляются очки
- наличие окончания игры.

**Задание 3.** Напишите программу, которая показывает в окне курс продажи и покупки валют.

Некоторые требования к программе:

- количество валют - не менее 5
- для каждой валюты должен быть представлен флаг страны и символ национальной валюты
- сведения о валютах программа должна брать с 3 разных сайтов и отображать среднее значение

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

Формой контроля знаний по дисциплине « Разработка кросс-платформенных приложений» являются две текущие аттестации в форме тестов и итоговая аттестация в формах защиты курсовой работы и экзамена.

Неделя текущей/промежуточного контроля	Вид оценочного средства	Код компетенций, оценивающий знания, умения, навыки	Содержание оценочного средства	Требования к выполнению	Срок сдачи (неделя семестра)	Критерии оценки по содержанию и качеству с указанием баллов
7-8 15-16	Тестирование	ПК-1, ПК-3, ПК-4, ПК-5, ПК-6, ПК-7	25 вопросов	Компьютерное тестирование. Время отведено на	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка - 0 Удовлетворительно - от 51% правильных ответов. Хорошо - от 70%.

				процедуру – 30 минут		Отлично – от 90%.
согласно КУГ	Защита курсовой работы	ПК-1, ПК-3, ПК-4, ПК-5, ПК-6, ПК-7	1 вопрос	Проводится в устной форме, путем ответа на вопросы. Время, отведенное на процедуру – 30 минут.	Результаты предоставляются в день проведения защиты курсовой работы	Критерии оценки: <b>«Отлично»:</b> <ul style="list-style-type: none"> <li>• знание основных понятий предмета;</li> <li>• умение использовать и применять полученные знания на практике;</li> <li>• работа на практических занятиях;</li> <li>• знание основных научных теорий, изучаемых предметов;</li> <li>• ответ на вопросы билета.</li> </ul> <b>«Хорошо»:</b> <ul style="list-style-type: none"> <li>• знание основных понятий предмета;</li> <li>• умение использовать и применять полученные знания на практике;</li> <li>• работа на практических занятиях;</li> <li>• знание основных научных теорий, изучаемых предметов;</li> <li>• ответы на вопросы билета неполные</li> </ul> <b>«Удовлетворительно»:</b> <ul style="list-style-type: none"> <li>• демонстрирует частичные знания по темам дисциплин;</li> <li>• незнание неумение использовать и применять полученные знания на практике;</li> <li>• не работал на практических занятиях;</li> </ul> <b>«Неудовлетворительно»:</b> <ul style="list-style-type: none"> <li>• демонстрирует частичные знания по темам дисциплин;</li> <li>• незнание основных понятий предмета;</li> <li>• неумение использовать и применять полученные знания на практике;</li> <li>• не работал на практических занятиях;</li> </ul> не отвечает на вопросы.
согласно КУГ	Экзамен	ПК-1, ПК-3, ПК-4, ПК-5, ПК-6, ПК-7	2 вопроса, 1 практическое задание	Экзамен проводится в 2 этапа: устной форме, путем ответа на вопросы и демонстрация результатов выполнения	Результаты предоставляются в день проведения экзамена	Критерии оценки: <b>«Отлично»:</b> <ul style="list-style-type: none"> <li>• знание основных понятий предмета;</li> <li>• умение использовать и применять полученные знания на практике;</li> <li>• работа на практических занятиях;</li> <li>• знание основных научных теорий, изучаемых предметов;</li> <li>• ответ на вопросы билета.</li> </ul> <b>«Хорошо»:</b> <ul style="list-style-type: none"> <li>• знание основных понятий предмета;</li> </ul>

				<p>ния практич еского задания. Время отведенн ое на процеду ру – 30 минут.</p>	<ul style="list-style-type: none"> <li>• умение использовать и применять полученные знания на практике;</li> <li>• работа на практических занятиях;</li> <li>• знание основных научных теорий, изучаемых предметов;</li> <li>• ответы на вопросы билета неполные</li> </ul> <p><b>«Удовлетворительно»:</b></p> <ul style="list-style-type: none"> <li>• демонстрирует частичные знания по темам дисциплин;</li> <li>• незнание неумение использовать и применять полученные знания на практике;</li> <li>• не работал на практических занятиях;</li> </ul> <p><b>«Неудовлетворительно»:</b></p> <ul style="list-style-type: none"> <li>• демонстрирует частичные знания по темам дисциплин;</li> <li>• незнание основных понятий предмета;</li> <li>• неумение использовать и применять полученные знания на практике;</li> <li>• не работал на практических занятиях;</li> <li>• не отвечает на вопросы.</li> </ul>
--	--	--	--	--	--

#### 4.1.Примерная тематика тестовых заданий:

##### 1 Кросс-платформенное приложение - это

- a) приложение для операционной системы Windows
- b) приложение для операционной системы MacOS
- c) приложение для операционной системы Linux
- d) все выше перечисленное.

##### 2 Язык программирования, позволяющий разрабатывать кросс-платформенные приложения:

- a) QBASIC;
- b) Haskell
- c) Python.

##### 3 Среда разработки для создания кросс-платформенных приложений:

- a) Visual Studio
- b) Delphi
- c) Idle
- d) Code:Blocks

##### 4 Графические библиотеки, имеющие реализацию для нескольких операционных систем:

- a) GTK
- b) QT
- c) TKInter
- d) ничего из выше перечисленного

##### 5 Язык программирования, работающий в современных браузерах:

- a) JavaScript
- b) C#
- c) Python
- d) ничего из выше перечисленного

**6 Язык разметки:**

- a) JavaScript
- b) Python
- c) C#
- d) HTML

**7 Каркас исполнения, позволяющий разрабатывать приложения с применением языка программирования**

JavaScript:

- a) NW.js
- b) node.js
- c) OpenGL
- d) DirectX

**8 Кроссплатформенная графическая библиотека для упрощения разработки трехмерных приложений:**

- a) NW.js
- b) node.js
- c) OpenGL
- d) DirectX

**9 Модуль каркаса исполнения NW.js для работы с файловой системой:**

- a) fs
- b) files
- c) files.js
- d) system

**10 Каскадные таблицы стилей:**

- a) CSS
- b) HTML
- c) JavaScript
- d) C#

#### **4.2. Примерная тематика заданий на курсовую работу**

Цель курсовой работы – практическое освоение методов проектирования мобильных приложений. При этом студенты должны выполнить следующее:

- 1) Сформулировать требования к разрабатываемому приложению
- 2) Выбрать мобильную ОС для разработки
- 3) Спроектировать пользовательский интерфейс приложения
- 4) Создать работающее приложение

**Примерные темы курсовых работ:**

- 1) История создания кроссплатформенных систем программирования.
- 2) Основные направления Java.
- 3) Среды разработки для Net.

- 4) Типы данных в Java, C#, C++, Python.
- 5) Абстракции и объекты.
- 6) Абстрактные классы.
- 7) Интерфейсы.
- 8) Управление доступом.
- 9) Инкапсуляция.
- 10) Наследование и полиморфизм.
- 11) Коллекции объектов.
- 12) Обработка ошибок и исключения.
- 13) Система ввода-вывода.
- 14) События и их обработка
- 15) Библиотека Swing.
- 16) Концепция Model-View-Controller.
- 17) Диспетчеры компоновки.
- 18) Библиотека Qt.
- 19) Библиотека wxWidgets.
- 20) Работа с сетевыми протоколами.
- 21) Интернационализация.
- 22) Работа с базами данных.
- 23) Работа со звуком и графикой.

#### **4.3. Типовые вопросы, выносимые на экзамен**

- 1) История создания кроссплатформенных систем программирования.
- 2) Основные направления Java.
- 3) Особенности языка программирования JavaScript.
- 4) Идеология разработки для платформы .NET.
- 5) Особенности языка программирования C#.
- 6) Сравнение типов данных в языках программирования Java, C#, C++, Python, JavaScript.
- 7) Особенности реализации парадигм объектно-ориентированного программирования.
- 8) Управление доступом.
- 9) Инкапсуляция.
- 10) Наследование и полиморфизм.
- 11) Коллекции объектов.
- 12) Обработка ошибок и исключения.
- 13) Внутренние и анонимные (безымянные) внутренние классы.
- 14) Система ввода-вывода.
- 15) События и их обработка
- 16) Библиотека для реализации графического интерфейса.
- 17) Диспетчеры компоновки.
- 18) Библиотека для реализации графического интерфейса: QT.
- 19) Библиотека для реализации графического интерфейса: wxWidgets.
- 20) Работа с сетевыми протоколами.
- 21) Интернационализация.

- 22) Работа с базами данных.
- 23) Работа со звуком и графикой.
- 24) Удаленный вызов методов.

Итоговое начисление баллов по дисциплине осуществляется в соответствии с разработанной и внедренной балльно-рейтинговой системой контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И  
ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО  
ОСВОЕНИЮ ДИСЦИПЛИНЫ  
«РАЗРАБОТКА КРОСС-ПЛАТФОРМЕННЫХ ПРИЛОЖЕНИЙ»**

**(Приложение 2 к рабочей программе)**

**Направление подготовки: 09.03.04 Программная инженерия**

**Профиль: проектирование и разработка программного обеспечения**

**Уровень высшего образования: бакалавриат**

**Форма обучения: очная**

Королев  
2023

## **1. Общие положения**

Целью дисциплины "Разработка кросс-платформенных приложений" является изучение основ разработки кроссплатформенных приложений, включающие в себя:

- Особенности разработки кроссплатформенных приложений;
- Принципы разработки кроссплатформенных приложений;
- Архитектуру и жизненный цикл кроссплатформенных приложений;
- Работу основных компонентов кроссплатформенных приложений;
- Технологии кроссплатформенной разработки;
- Различные библиотеки и фреймворки, применяемые при кросс-платформенной разработке.

Задача дисциплины - получить обучающимся необходимый набор знаний и навыков для дальнейшей работы в области кроссплатформенной разработки и более глубокого изучения данной отрасли программирования.

## **2. Указания по проведению практических (семинарских) занятий**

### **Практическая работа 1.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 1. Введение в кросс-платформенное программирование.

Продолжительность занятия 4 часа.

### **Практическая работа 2.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 2. Разработка программы на основе объектно-ориентированного подхода в Java.

Продолжительность занятия 4 часа

### **Практическая работа 3.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 3. Организация сетевого взаимодействия. Организация взаимодействия с помощью графического интерфейса.

Продолжительность занятия 4 часа.

### **Практическая работа 4.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Разработка серверных приложений. Работа по протоколу http/https.

Продолжительность занятия 4 часа.

### **Практическая работа 5.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 5. Паттерны разработки мобильного программного обеспечения.

Продолжительность занятия 4 часа.

### **Практическая работа 6.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 6. Проектирование GUI. Библиотеки и фреймворки.

Продолжительность занятия 4 часа

### **Практическая работа 7.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 7 Сервлеты. Серверные страницы.

Продолжительность занятия 4 часа.

### **Практическая работа 8.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 8. Многопоточность. Асинхронность.

Продолжительность занятия 4 часа.

## **3. Указания по проведению лабораторного практикума**

Не предусмотрено учебным планом

## **4. Указания по проведению самостоятельной работы студентов**

Самостоятельная работа обучающихся должна строиться в соответствии со следующими документами:

- Методическими указаниями по организации самостоятельной работы по дисциплине «Разработка кросс-платформенных приложений».
- Лекциями по дисциплине «Разработка кросс-платформенных приложений» [размещены в формате для чтения на образовательном портале Университета].
- Методическими рекомендациями к выполнению практических работ по дисциплине «Разработка кросс-платформенных приложений» [размещены в формате для чтения на образовательном портале Университета].

Для успешного освоения дисциплины, необходимо самостоятельно детально изучить рекомендуемые источники информации.

## 5. Перечень основной и дополнительной учебной литературы

### Основная литература:

1. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения. Часть 1: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова; Южный федеральный университет. - Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2019. - 139 с. - ISBN 978-5-9275-3367-1. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1088203>. - Режим доступа: по подписке.

2. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения. Часть 2: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова; Южный федеральный университет. - Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2019. - 168 с. - ISBN 978-5-9275-3368-8. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1088205>. - Режим доступа: по подписке.

3. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения: в 3 ч. Часть 3: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова; Южный федеральный университет. - Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2021. - 214 с. - ISBN 78-5-9275-3628-3. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1894418>. - Режим доступа: по подписке.

4. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке C++: учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев ; под ред. Л.Г. Гагариной. - Москва: ФОРУМ: ИНФРА-М, 2021. - 512 с. + Доп. материалы [Электронный ресурс]. - (Среднее профессиональное образование). - ISBN 978-5-8199-0699-6. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1172261>. - Режим доступа: по подписке.

### Дополнительная литература:

1. Кузин, А. В. Программирование на языке Си: учебное пособие / А.В. Кузин, Е.В. Чумакова. - Москва: ФОРУМ: ИНФРА-М, 2023. - 143 с. - (Среднее профессиональное образование). - ISBN 978-5-00091-556-1. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1878382>. - Режим доступа: по подписке.

2. Кузин, А. В. Программирование на языке Си: учебное пособие / А.В. Кузин, Е.В. Чумакова. - Москва: ФОРУМ: ИНФРА-М, 2021. - 144 с. - (Высшее образование). - ISBN 978-5-00091-066-5. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1222078>. - Режим доступа: по подписке.

3. Бедердинова, О. И. Программирование на языках высокого уровня: учеб. пособие / О.И. Бедердинова, Т.А. Минеева, Ю.А. Водовозова. - Москва: ИНФРА-

М, 2019. - 159 с. - Текст: электронный. - URL:  
<https://znanium.com/catalog/product/1044396>. - Режим доступа: по подписке.

## **6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

### **Интернет-ресурсы:**

1. <http://www.ru.wikipedia.org>
2. <http://intuit.ru>
3. <http://metanit.com>
4. <http://developer.android.com>
5. <https://biblioclub.ru/> - Университетская библиотека ONLINE.
6. <https://www.java.com/ru/> - Сайт разработчика Java
7. <https://www.rucont.ru/> - Электронная библиотека Руконт.
8. <https://znanium.com/> - Электронная библиотека издательства Инфра-М.
9. <https://e.lanbook.com/> - Электронная библиотека Лань.
10. <https://www.jetbrains.com/ru-ru/idea/> - Сайт разработчика IntelliJ IDEA.
11. <https://github.com/> - Веб-сервис для хостинга IT-проектов и их совместной разработки.
12. <https://www.eclipse.org/> - Сайт разработчика Eclipse
13. <https://jug.ru/> - Сайт Java-разработчиков

### **Перечень программного обеспечения:**

MSOffice, Double Commander, VS Code, Eclipse IDE, IntelliJ IDEA, JDK Java SE8 и выше, Notepad++, Android Studio, Open Server, Mozilla Firefox, Google Chrome, Adobe Reader XI или Adobe Acrobat Reader DC, Kaspersky Endpoint Security для Windows.

### **Информационные справочные системы:**

1. Электронные ресурсы образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Разработка кросс-платформенных приложений».