

**Общество с ограниченной ответственностью «ЯРУС»  
(ООО «ЯРУС»)**

---

УТВЕРЖДАЮ

Генеральный директор

ООО «ЯРУС»

Л.В. Визирева

(приказ № 2 от «21» апреля 2026 г.)

**ПРОГРАММА ПРОФЕССИОНАЛЬНОГО ОБУЧЕНИЯ –  
ПРОГРАММА ПРОФЕССИОНАЛЬНОЙ ПОДГОТОВКИ**

*«Коммерческая разработка»*

*132 академических часа*

Вид профессиональной деятельности: Разработка компьютерного программного обеспечения.

Наименование присваиваемой квалификации: «Программист».

г. Всеволожск, 2026

**СОДЕРЖАНИЕ**

1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ.....	3
1.1 Общие положения.....	3
1.2 Цель освоения и характеристика получаемой квалификации.....	4
1.3 Планируемые результаты обучения.....	5
1.4 Учебно-тематический план.....	15
1.5 Календарный учебный график.....	17
1.6 Рабочие программы модулей.....	19
1.7 Организационно-педагогические условия.....	24
1.8 Формы аттестации.....	26
2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ.....	27
2.1. Текущий контроль успеваемости.....	27
2.2. Промежуточная аттестация.....	28
2.3. Итоговая аттестация.....	28

## **1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ**

### **1.1 Общие положения**

#### **1.1.1 Нормативные правовые основания разработки программы**

Нормативные правовые основания для разработки программы профессионального обучения – программы профессиональной подготовки «*Коммерческая разработка*» (далее допускается – «Программа») составляют:

- Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Минпросвещения России от 26.08.2020 N 438 «Об утверждении Порядка организации и осуществления образовательной деятельности по основным программам профессионального обучения»;
- Приказ Минтруда России от 20.07.2022 N 424н "Об утверждении профессионального стандарта "Программист" (Зарегистрировано в Минюсте России 22.08.2022 N 69720).

Программа разработана на основе профессионального стандарта: «Программист» (утвержден Приказом Минтруда России № 424н от 20.07.2022).

Программа разработана на основе установленных квалификационных требований по должностям «Инженер-программист (программист)», «Техник-программист».

Программа разработана на основе требований федерального государственного образовательного стандарта среднего профессионального образования 09.00.00 «Информатика и вычислительная техника» к результатам освоения образовательных программ.

Программа разработана на основе требований федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.00.00 «Информатика и вычислительная техника» к результатам освоения образовательных программ.

#### **1.1.2 Требования к обучающимся**

Требования к уровню профессионального образования: к освоению Программы допускаются лица, имеющие общее среднее образование, ранее не имевших профессии рабочего или должности служащего.

#### **1.1.3 Особенности адаптации образовательной программы для лиц с ограниченными возможностями здоровья**

Разработка адаптированной образовательной программы для лиц с ОВЗ и/или инвалидностью или обновление уже существующей образовательной программы определяются индивидуальной программой реабилитации инвалида (при наличии), рекомендациями заключения ПМПК (при наличии) и осуществляются по заявлению слушателя (законного представителя).

**1.1.4 Форма обучения:** заочная с применением исключительно электронного обучения и дистанционно-образовательных технологий.

**1.1.5 Трудоемкость освоения:** 132 академических часа, включая все виды контактной и самостоятельной работы обучающегося.

**1.1.6 Срок освоения:** 32 календарных дня (16 недель).

**1.1.7 Форма документа, выдаваемого по результатам освоения программы:** лицам, успешно освоившим программу профессионального обучения и прошедшим итоговую аттестацию, выдается свидетельство о профессии рабочего, должности служащего установленного образца.

## **1.2 Цель освоения и характеристика получаемой квалификации**

### **1.2.1 Цель освоения**

Целью освоения программы является формирование компетенций, определенных профессиональным стандартом «Программист» в целях разработки, отладки, проверки работоспособности, модификации компьютерного программного обеспечения.

### **1.2.2 Характеристика получаемой квалификации и связанных с ней видов профессиональной деятельности, трудовых функций и (или) уровней квалификации**

Вид профессиональной деятельности: Разработка компьютерного программного обеспечения.

Обобщенные (конкретные) трудовые функции, подлежащие освоению:

A/01.3 Формализация и алгоритмизация поставленных задач для разработки программного кода.

A/02.3 Написание программного кода с использованием языков программирования, определения и манипулирования данными в базах данных.

A/03.3 Оформление программного кода в соответствии с установленными требованиями.

A/04.3 Работа с системой управления версиями программного кода.

A/05.3 Проверка и отладка программного кода.

B/01.4 Разработка процедур проверки работоспособности и измерения характеристик компьютерного программного обеспечения.

B/02.4 Разработка тестовых наборов данных для проверки работоспособности компьютерного программного обеспечения.

B/03.4 Проверка работоспособности компьютерного программного обеспечения.

B/04.4 Рефакторинг, оптимизация и инспекция программного кода.

B/05.4 Исправление дефектов программного кода, зафиксированных в базе данных дефектов.

B/06.4 Осуществление сборки однородных программных модулей в программный проект.

Уровень квалификации в соответствии с профессиональным стандартом: 4 уровень.

### 1.3 Планируемые результаты обучения

Программа направлена на получение компетенции, необходимой для выполнения профессиональной деятельности в области разработки компьютерного программного обеспечения, приобретение квалификации «Программист».

Выпускник, окончивший полный курс обучения по Программе, должен обладать следующими общепрофессиональными компетенциями (ОПК, ПСК<sup>1</sup>) и готов выполнять следующие трудовые функции в соответствии с профессиональным стандартом 4 «Программист».

Таблица 1 – Получаемые компетенции

Код компетенций	Компетенции	Код трудовых функций	Трудовые функции
ОПК-1	Способен использовать прикладные знания и программное обеспечение при решении профессиональных задач	A/03.3 (ПСК-1.2)	Оформление программного кода в соответствии с установленными требованиями
ОПК-2	Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности	A/01.3 (ПСК-1.1)	Формализация и алгоритмизация поставленных задач для разработки программного кода
		A/04.3 (ПСК-2.2)	Работа с системой управления версиями программного кода
ОПК-3	Способен установить программное и аппаратное обеспечение информационных и автоматизированных систем	B/06.4 (ПСК-2.6)	Осуществление сборки однородных программных модулей в программный проект
ОПК-4	Способен участвовать в настройке и наладке программно-аппаратных комплексов	A/05.3 (ПСК-2.3)	Проверка и отладка программного кода
		B/01.4 (ПСК-1.3)	Разработка процедур проверки работоспособности и измерения характеристик компьютерного программного обеспечения
		B/02.4 (ПСК-1.4)	Разработка тестовых наборов данных для проверки

<sup>1</sup> ПСК – профессионально-специализированные компетенции

Код компетенций	Компетенции	Код трудовых функций	Трудовые функции
			работоспособности компьютерного программного обеспечения
		В/03.4 (ПСК-1.5)	Проверка работоспособности компьютерного программного обеспечения
		В/04.4 (ПСК-2.4)	Рефакторинг, оптимизация и инспекция программного кода
		В/05.4 (ПСК-2.5)	Исправление дефектов программного кода, зафиксированных в базе данных дефектов
ОПК-5	Способен разрабатывать алгоритмы и программы, пригодные для практического применения	А/02.3 (ПСК-2.1)	Написание программного кода с использованием языков программирования, определения и манипулирования данными в базах данных
		В/06.4 (ПСК-2.6)	Осуществление сборки однородных программных модулей в программный проект

Таблица 2 – Планируемые результаты обучения

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
ВД 1. Организационно-техническая деятельность	ПСК 1.1 Формализация и алгоритмизация поставленных задач для разработки программного кода	<p>Методы и приемы формализации поставленных задач</p> <p>Языки формализации функциональных спецификаций</p> <p>Методы и приемы алгоритмизации поставленных задач</p> <p>Нотации и программное обеспечение для графического отображения алгоритмов</p> <p>Алгоритмы решения типичных задач, области и способы их применения</p>	<p>Использовать методы и приемы формализации поставленных задач</p> <p>Использовать методы и приемы алгоритмизации поставленных задач</p> <p>Использовать программное обеспечение для графического отображения алгоритмов</p> <p>Применять алгоритмы решения типовых задач в соответствующих областях</p> <p>Осуществлять коммуникации с заинтересованными сторонами</p>	<p>Составление формализованных описаний решений поставленных задач в соответствии с требованиями технического задания или внутренних документов организации</p> <p>Разработка алгоритмов решения поставленных задач в соответствии с требованиями технического задания или внутренних документов организации</p> <p>Проверка корректности алгоритмов решения поставленных задач</p> <p>Оценка и согласование сроков выполнения поставленных задач</p>
	ПСК 1.2 Оформление программного кода в соответствии с установленными требованиями	<p>Инструментарий для создания и актуализации исходных текстов программ</p> <p>Методы повышения читаемости программного кода</p>	<p>Применять нормативно-технические документы (стандарты и регламенты), определяющие требования к оформлению программного кода</p> <p>Применять инструментарий для</p>	<p>Приведение наименований переменных, функций, классов, структур данных и файлов в соответствие с нормативно-техническими документами (стандартами и регламентами),</p>

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
		<p>Системы кодировки символов, форматы хранения исходных текстов программ</p> <p>Нормативно-технические документы (стандарты и регламенты), определяющие требования к оформлению программного кода</p> <p>Основные стандарты оформления технической документации на компьютерное программное обеспечение</p>	<p>создания и актуализации исходных текстов программ</p> <p>Применять заданные стандарты и шаблоны для составления и оформления технической документации</p> <p>Осуществлять коммуникации с заинтересованными сторонами</p>	<p>определяющими требованиями к оформлению программного кода</p> <p>Структурирование исходного программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода</p>
	ПСК 1.3 Разработка процедур проверки работоспособности и измерения характеристик компьютерного программного обеспечения	<p>Методы автоматической и автоматизированной проверки работоспособности компьютерного программного обеспечения</p> <p>Основные виды диагностических данных проверки работоспособности компьютерного программного обеспечения и способы их представления</p> <p>Языки, утилиты и среды программирования и средства пакетного выполнения процедур</p>	<p>Писать программный код процедур проверки работоспособности компьютерного программного обеспечения на выбранном языке программирования</p> <p>Использовать выбранную среду программирования для разработки процедур проверки работоспособности компьютерного программного обеспечения</p> <p>Применять заданные стандарты и шаблоны для составления и оформления технической документации</p>	<p>Разработка процедуры проверки работоспособности компьютерного программного обеспечения</p> <p>Разработка процедуры сбора диагностических данных проверки работоспособности компьютерного программного обеспечения</p> <p>Разработка процедуры измерения требуемых характеристик компьютерного</p>

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
		<p>Типовые метрики компьютерного программного обеспечения</p> <p>Основные методы измерения и оценки характеристик компьютерного программного обеспечения</p> <p>Основные стандарты оформления технической документации на компьютерное программное обеспечение</p>	<p>Осуществлять коммуникации с заинтересованными сторонами</p>	<p>программного обеспечения</p> <p>Оформление технической документации на компьютерное программное обеспечение по заданному стандарту или шаблону</p>
	<p>ПСК 1.4</p> <p>Разработка тестовых наборов данных для проверки работоспособности компьютерного программного обеспечения</p>	<p>Методы создания и документирования контрольных примеров и тестовых наборов данных</p> <p>Правила, алгоритмы и технологии создания тестовых наборов данных</p> <p>Требования к структуре и форматам хранения тестовых наборов данных</p>	<p>Проводить переговоры и совещания по тематике технологий больших данных с целью формирования и согласования технического задания на создание методической и технологической инфраструктуры больших данных</p> <p>Проводить презентации проектов информационно-технологической инфраструктуры и методической и технологической инфраструктуры технологий больших данных для заинтересованных лиц и (или) представителями подразделений организации с целью формирования и согласования технического задания на создание методической и технологической инфраструктуры технологий больших данных</p>	<p>Постановка целей создания методической и технологической инфраструктуры больших данных</p> <p>Разработка концепции методической и технологической инфраструктуры больших данных</p> <p>Оценка современных методов и инструментальных средств анализа больших данных</p> <p>Разработка технического задания на создание методической и технологической</p>

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
			Организовывать разработку и согласование технического задания	инфраструктуры больших данных
	ПСК 1.5 Проверка работоспособности компьютерного программного обеспечения	<p>Методы и средства проверки работоспособности компьютерного программного обеспечения</p> <p>Среды проверки работоспособности и отладки компьютерного программного обеспечения</p> <p>Государственные стандарты испытания автоматизированных систем</p> <p>Руководящие документы по стандартизации требований к документам автоматизированных систем</p>	<p>Применять методы и средства проверки работоспособности компьютерного программного обеспечения</p> <p>Интерпретировать диагностические данные проверки работоспособности компьютерного программного обеспечения</p> <p>Анализировать значения полученных характеристик компьютерного программного обеспечения</p> <p>Документировать результаты проверки работоспособности компьютерного программного обеспечения</p> <p>Осуществлять коммуникации с заинтересованными сторонами</p>	<p>Проверка работоспособности компьютерного программного обеспечения на основе разработанных тестовых наборов данных</p> <p>Оценка соответствия компьютерного программного обеспечения требуемым характеристикам</p> <p>Сбор и анализ полученных результатов проверки работоспособности компьютерного программного обеспечения</p>
ВД 2. Аналитическая прикладная деятельность	ПСК 2.1 Написание программного кода с использованием языков	Синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	<p>Применять выбранные языки программирования для написания программного кода</p> <p>Использовать выбранную среду программирования и средства</p>	Создание программного кода в соответствии с техническим заданием (готовыми спецификациями)

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
	программирования, определения и манипулирования данными в базах данных	<p>Методологии разработки компьютерного программного обеспечения</p> <p>Методологии и технологии проектирования и использования баз данных</p> <p>Технологии программирования</p> <p>Особенности выбранной среды программирования и системы управления базами данных</p> <p>Компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними</p>	<p>системы управления базами данных</p> <p>Использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода</p>	<p>Оптимизация программного кода с использованием специализированных программных средств</p> <p>Оценка и согласование сроков выполнения поставленных задач</p>
	ПСК 2.2 Работа с системой управления версиями программного кода	<p>Возможности используемой системы управления версиями и вспомогательных инструментальных программных средств</p> <p>Установленный регламент использования системы управления версиями</p>	<p>Использовать выбранную систему управления версиями</p> <p>Использовать вспомогательные инструментальные программные средства для обработки исходного текста программного кода</p> <p>Выполнять действия, соответствующие установленному регламенту используемой системы управления версиями</p>	<p>Регистрация изменений исходного текста программного кода в системе управления версиями</p> <p>Слияние, разделение и сравнение исходных текстов программного кода</p> <p>Сохранение сделанных изменений программного кода</p>

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
				кода в соответствии с регламентом управления версиями
	ПСК 2.3 Проверка и отладка программного кода	<p>Методы и приемы отладки программного кода</p> <p>Типы и форматы сообщений об ошибках, предупреждений</p> <p>Способы использования технологических журналов, форматы и типы записей журналов</p> <p>Современные компиляторы, отладчики и оптимизаторы программного кода</p> <p>Сообщения о состоянии аппаратных средств</p>	<p>Выявлять ошибки в программном коде</p> <p>Применять методы и приемы отладки программного кода</p> <p>Интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов</p> <p>Применять современные компиляторы, отладчики и оптимизаторы программного кода</p>	<p>Анализ и проверка исходного программного кода</p> <p>Отладка программного кода на уровне программных модулей</p> <p>Отладка программного кода на уровне межмодульных взаимодействий и взаимодействий с окружением</p>
	ПСК 2.4 Рефакторинг, оптимизация и инспекция программного кода	<p>Методы и средства рефакторинга, оптимизации и инспекции программного кода</p> <p>Языки программирования и среды разработки</p> <p>Нормативно-технические документы (стандарты и регламенты), регламентирующие требования к программному коду, порядок отражения изменений в</p>	<p>Применять методы, средства рефакторинга, оптимизации и инспекции программного кода</p> <p>Применять инструментальные средства коллективной работы над программным кодом</p> <p>Публиковать результаты рефакторинга, оптимизации и инспекции в коллективной базе знаний</p>	<p>Анализ программного кода на соответствие требованиям по читаемости и производительности</p> <p>Инспекция программного кода для поиска не обнаруженных на ранних стадиях разработки компьютерного программного обеспечения ошибок и критических мест</p>

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
		системе управления версиями, порядок отражения результатов рефакторинга, оптимизации и инспекции в коллективной базе знаний	Использовать систему управления версиями для регистрации произведенных изменений	Внесение изменений в программный код и проверка его работоспособности
	ПСК 2.5 Исправление дефектов программного кода, зафиксированных в базе данных дефектов	<p>Методы и приемы отладки программного кода</p> <p>Типичные ошибки, возникающие при разработке компьютерного программного обеспечения, методы их диагностики и исправления</p>	<p>Применять методы и приемы отладки дефектного программного кода</p> <p>Интерпретировать сообщения, предупреждения, записи технологических журналов об ошибках, возникающих при выполнении дефектного кода</p>	<p>Воспроизведение дефектов программного кода, зафиксированных в базе данных дефектов</p> <p>Установление причин возникновения дефектов программного кода</p> <p>Внесение изменений в программный код для устранения выявленных дефектов</p>
	ПСК 2.6 Осуществление сборки однородных программных модулей в программный проект	<p>Методы и средства сборки модулей в программный проект в средах разработки компьютерного программного обеспечения</p> <p>Возможности настройки программного проекта в средах разработки компьютерного программного обеспечения</p> <p>Интерфейсы взаимодействия внутренних модулей программного проекта</p>	<p>Выполнять процедуры сборки однородных (однойязыковых) программных модулей в программный проект в средах разработки компьютерного программного обеспечения</p> <p>Производить настройки параметров программного проекта и осуществлять запуск процедур сборки</p> <p>Проводить проверку</p>	<p>Сборка однородных (однойязыковых) программных модулей в программный проект в средах разработки компьютерного программного обеспечения</p> <p>Проверка работоспособности собранного программного проекта</p>

Виды деятельности	Код и наименование компетенции	Показатели освоения компетенции		
		Знания	Умения	Практический опыт
		<p>Методы и средства проверки работоспособности программных проектов</p> <p>Языки, утилиты и среды программирования, средства пакетного выполнения процедур</p>	<p>работоспособности программного проекта</p> <p>Документировать произведенные действия, выявленные проблемы и способы их устранения</p> <p>Создавать резервные копии программного проекта и данных, выполнять восстановление, обеспечивать целостность программного проекта и данных</p>	<p>Внесение изменений в процедуру сборки однородных (одноязыковых) программных модулей в программный проект</p>

## 1.4 Учебно-тематический план

Таблица 3 – Учебно-тематический план

Наименование модулей, тем, вида аттестации	Трудоемкость, ак. час				Формы аттестации
	Итого	Виды занятий		Самостоятельная работа	
		Лекции	Практические занятия		
<b>Модуль 1: Процедурное программирование</b>	<b>30</b>	<b>7</b>	<b>9</b>	<b>14</b>	-
Тема 1.1: Начало работы со средой разработки	2	1	1	-	Решение практических задач
Тема 1.2: Работа с переменными	4	1	1	2	Решение практических задач
Тема 1.3: Работа с условными конструкциями	4	1	1	2	Решение практических задач
Тема 1.4: Работа с циклическими конструкциями	5	1	2	2	Решение практических задач
Тема 1.5: Изучение подходов к организации программного кода	4	1	1	2	Решение практических задач
Тема 1.6: Работа с коллекциями	7	1	2	4	Решение практических задач
Тема 1.7: Работа с системой контроля версий	4	1	1	2	Решение практических задач
<b>Промежуточная аттестация</b>	-	-	-	-	Выполнение практической работы (проекта)
<b>Модуль 2: Объектно-Ориентированное программирование</b>	<b>39</b>	<b>7</b>	<b>11</b>	<b>21</b>	-
Тема 2.1: Объектно-ориентированное программирование	7	1	2	4	Решение практических задач
Тема 2.2: Продвинутая работа с системой контроля версий	4	1	1	2	Решение практических задач
Тема 2.3: Основы командной разработки	4	1	1	2	Решение практических задач
Тема 2.4: Основные подходы к проектированию приложений. Анализ постановки задачи и основных требований	7	1	2	4	Решение практических задач
Тема 2.5: Изучение архитектуры приложений. Паттерны проектирования	6	1	2	3	Решение практических задач
Тема 2.6: Работа с библиотекой Telegram.Bot	6	1	1	4	Решение практических задач
Тема 2.7: Основы модульного тестирования приложений	5	1	2	2	Решение практических задач
<b>Промежуточная аттестация</b>	-	-	-	-	Выполнение практической работы (проекта)
<b>Модуль 3: Командная разработка коммерческих приложений</b>	<b>61</b>	<b>11</b>	<b>17</b>	<b>33</b>	-
Тема 3.1: Основы устройства реляционных баз данных	8	1	2	5	Решение практических задач

Наименование модулей, тем, вида аттестации	Трудоемкость, ак. час				Формы аттестации
	Итого	Виды занятий		Самостоятельная работа	
		Лекции	Практические занятия		
Тема 3.2: Нормализация баз данных	3	1	1	1	Решение практических задач
Тема 3.3: Написание запросов на языке SQL	3	1	1	1	Решение практических задач
Тема 3.4: Способы связи приложения с базами данных	6	1	2	3	Решение практических задач
Тема 3.5: Реализация бизнес-логики приложения	9	1	2	6	Решение практических задач
Тема 3.6: Изучение архитектуры приложений. Организация слоя для работы с БД	4	1	1	2	Решение практических задач
Тема 3.7: Изучение архитектуры приложений. Организация бизнес-логики	7	1	2	4	Решение практических задач
Тема 3.8: Подходы к визуализации информации. Основы HTML и CSS	6	1	1	4	Решение практических задач
Тема 3.9: Работа с Blazor-проектом	3	1	1	1	Решение практических задач
Тема 3.10: Изучение архитектуры приложений. Layout Pattern	5	1	2	2	Решение практических задач
Тема 3.11: Основы реализации автотестов. SeleniumWebDriver	7	1	2	4	Решение практических задач
<i>Промежуточная аттестация</i>	-	-	-	-	Выполнение практической работы (проекта)
<b>ВСЕГО</b>	<b>130</b>	<b>25</b>	<b>37</b>	<b>68</b>	-
<b>Итоговая аттестация – квалификационный экзамен</b>	<b>2</b>	<b>2</b>	-	-	Защита итоговой практической работы (проекта)
<b>Общая трудоемкость программы</b>	<b>132</b>	<b>27</b>	<b>37</b>	<b>68</b>	-



Наименование модулей и тем	18 день	19 день	20 день	21 день	22 день	23 день	24 день	25 день	26 день	27 день	28 день	29 день	30 день	31 день	32 день	СР	ИА	Всего
<b>Модуль 3: Командная разработка коммерческих приложений</b>																		
Тема 3.1: Основы устройства реляционных баз данных	2			1												5		8
Тема 3.2: Нормализация баз данных		2														1		3
Тема 3.3: Написание запросов на языке SQL			2													1		3
Тема 3.4: Способы связи приложения с базами данных				1	2											3		6
Тема 3.5: Реализация бизнес-логики приложения						2		1								6		9
Тема 3.6: Изучение архитектуры приложений. Организация слоя для работы с БД							2									2		4
Тема 3.7: Изучение архитектуры приложений. Организация бизнес-логики								1	2							4		7
Тема 3.8: Подходы к визуализации информации. Основы HTML и CSS										2						4		6
Тема 3.9: Работа с Blazor-проектом											2					1		3
Тема 3.10: Изучение архитектуры приложений. Layout Pattern												2	1			2		5
Тема 3.11: Основы реализации автотестов. SeleniumWebDriver													1	2		4		7
Квалификационный экзамен															2		2	
<b>Итого за 18-32 день – 63 часа</b>																	<b>2</b>	<b>61</b>
<b>ИТОГО 132 часа</b>																	<b>2</b>	<b>130</b>

Условные обозначения	
СР	самостоятельная работа
ИА	итоговая аттестация

## 1.6 Рабочие программы модулей

Таблица 5 – Рабочая программа модуля

Наименование тем	Виды учебных занятий	ак. час	Содержание
<b>Модуль 1: Процедурное программирование</b>			
Тема 1.1: Начало работы со средой разработки	Л	1	<ul style="list-style-type: none"> <li>▪ Как установить</li> <li>▪ Какие пакеты выбрать</li> <li>▪ Интерфейс</li> <li>▪ Концепция проектов</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Настройка среды разработки</li> <li>▪ Создание проектов с необходимой структурой</li> </ul>
Тема 1.2: Работа с переменными	Л	1	<ul style="list-style-type: none"> <li>▪ Типы данных</li> <li>▪ Концепция переменной</li> <li>▪ Правила нейминга</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Операторы работы с переменными</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: Использование переменных</li> </ul>
Тема 1.3: Работа с условными конструкциями	Л	1	<ul style="list-style-type: none"> <li>▪ Логические операторы и типы</li> <li>▪ Алгебра логики</li> <li>▪ Концепция условных конструкций</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Использование условных конструкций</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: Использование условных конструкций</li> </ul>
Тема 1.4: Работа с циклическими конструкциями	Л	1	<ul style="list-style-type: none"> <li>▪ Концепция циклов</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Использование итерационного цикла</li> <li>▪ Использование циклов с условиями</li> <li>▪ Использование вложенных циклов</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: Использование циклов</li> </ul>
Тема 1.5: Изучение подходов к организации программного кода	Л	1	<ul style="list-style-type: none"> <li>▪ Концепция чёрного ящика</li> <li>▪ Параметры и возвращаемые значения методов</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Модернизация кода под использование методов</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: Использование методов</li> </ul>
Тема 1.6: Работа с	Л	1	<ul style="list-style-type: none"> <li>▪ Концепция коллекций</li> </ul>

Наименование тем	Виды учебных занятий	ак. час	Содержание
коллекциями	ПЗ	2	▪ Алгоритмы работы с коллекциями
	СР	4	▪ Задание: Использование коллекций
Тема 1.7: Работа с системой контроля версий	Л	1	▪ Концепция системы контроля версий
	ПЗ	1	▪ Практическая работа с Git
	СР	2	▪ Задание: Работа с системой контроля версий
<b>Модуль 2: Объектно-Ориентированное программирование</b>			
Тема 2.1: Объектно-ориентированное программирование	Л	1	▪ Концепция ООП ▪ Инкапсуляция, наследование, полиморфизм
	ПЗ	2	▪ Применение ООП на практике
	СР	4	▪ Задание: Объектно-ориентированное программирование
Тема 2.2: Продвинутая работа с системой контроля версий	Л	1	▪ Работа с ветками в Git
	ПЗ	1	▪ Применение стратегий бранчинга
	СР	2	▪ Задание: Продвинутая работа с Git
Тема 2.3: Основы командной разработки	Л	1	▪ Способы организации командной работы ▪ Работа с Jira/Trello
	ПЗ	1	▪ Применение практик командной работы в текущем проекте
	СР	2	▪ Задание: Основы командной разработки
Тема 2.4: Основные подходы к проектированию приложений. Анализ постановки задачи и основных требований	Л	1	▪ Концепция постановки задачи от заказчика ▪ Разбор типичных формулировок требований ▪ Техники декомпозиции: разбиение сложной задачи на подзадачи ▪ Выделение сущностей (Entities) и их атрибутов ▪ Определение взаимосвязей между сущностями ▪ Техники документирования требований
	ПЗ	2	▪ Анализ реального ТЗ

Наименование тем	Виды учебных занятий	ак. час	Содержание
	СР	4	<ul style="list-style-type: none"> <li>▪ Задание: Основные подходы к проектированию приложений. Анализ постановки задачи и основных требований</li> </ul>
Тема 2.5: Изучение архитектуры приложений. Паттерны проектирования	Л	1	<ul style="list-style-type: none"> <li>▪ Типы архитектур приложений (монолитная, двуслойная, многослойная, микросервисная)</li> <li>▪ Чистая архитектура</li> <li>▪ Репозиторий и Unit of Work</li> <li>▪ Паттерны GoF</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Анализ архитектуры существующего проекта</li> <li>▪ Рефакторинг с применением паттернов</li> </ul>
	СР	3	<ul style="list-style-type: none"> <li>▪ Задание: Изучение архитектуры приложений. Паттерны проектирования</li> </ul>
Тема 2.6: Работа с библиотекой Telegram.Bot	Л	1	<ul style="list-style-type: none"> <li>▪ Обзор возможностей Telegram Bot API</li> <li>▪ Установка и настройка библиотеки Telegram.Bot</li> <li>▪ Базовая архитектура Telegram-бота</li> <li>▪ Обработка сообщений и команд</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Создание простого бота-эхо</li> <li>▪ Создание бота-опросника</li> </ul>
	СР	4	<ul style="list-style-type: none"> <li>▪ Задание: Работа с библиотекой Telegram.Bot</li> </ul>
Тема 2.7: Основы модульного тестирования приложений	Л	1	<ul style="list-style-type: none"> <li>▪ Концепция тестирования приложений</li> <li>▪ Виды тестирования</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Написание юнит-тестов для программы</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: модульное тестирование</li> </ul>
<b>Модуль 3: Командная разработка коммерческих приложений</b>			
Тема 3.1: Основы устройства реляционных баз данных	Л	1	<ul style="list-style-type: none"> <li>▪ Введение в реляционные базы данных</li> <li>▪ Реляционная модель данных</li> <li>▪ Преимущества реляционных БД</li> <li>▪ Типы данных в SQL</li> <li>▪ Типы связей между таблицами</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Проектирование БД для интернет-магазина</li> </ul>
	СР	5	<ul style="list-style-type: none"> <li>▪ Задание: Основы устройства</li> </ul>

Наименование тем	Виды учебных занятий	ак. час	Содержание
			реляционных баз данных
Тема 3.2: Нормализация баз данных	Л	1	<ul style="list-style-type: none"> <li>▪ Зачем нужна нормализация</li> <li>▪ Нормальные формы</li> <li>▪ Денормализация</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Анализ и исправление плохой схемы БД</li> </ul>
	СР	1	<ul style="list-style-type: none"> <li>▪ Задание: Нормализация баз данных</li> </ul>
Тема 3.3: Написание запросов на языке SQL	Л	1	<ul style="list-style-type: none"> <li>▪ Основы SQL - CRUD операции</li> <li>▪ Агрегация данных и группировка</li> <li>▪ Соединение таблиц (JOIN)</li> <li>▪ Типы данных в SQL</li> <li>▪ Типы связей между таблицами</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Работа с БД университета</li> <li>▪ Решение реальных бизнес-задач</li> </ul>
	СР	1	<ul style="list-style-type: none"> <li>▪ Задание: Написание запросов на языке SQL</li> </ul>
Тема 3.4: Способы связи приложения с базами данных	Л	1	<ul style="list-style-type: none"> <li>▪ Архитектура доступа к данным</li> <li>▪ Основы работы с Dapper</li> <li>▪ Маппинг результатов</li> <li>▪ Работа с хранимыми процедурами</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Создание DAL для системы университета</li> <li>▪ Оптимизация производительности</li> </ul>
	СР	3	<ul style="list-style-type: none"> <li>▪ Задание: Способы связи приложения с базами данных</li> </ul>
Тема 3.5: Реализация бизнес-логики приложения	Л	1	<ul style="list-style-type: none"> <li>▪ Концепция бизнес-логики</li> <li>▪ Концепция Middleware</li> <li>▪ Обработка исключений</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Реализация глобального обработчика ошибок</li> <li>▪ Проработка бизнес-логики проекта</li> </ul>
	СР	6	<ul style="list-style-type: none"> <li>▪ Задание: Реализация бизнес-логики приложения</li> </ul>
Тема 3.6: Изучение архитектуры приложений. Организация слоя для работы с БД	Л	1	<ul style="list-style-type: none"> <li>▪ Архитектурные паттерны для работы с данными</li> <li>▪ Организация слоя данных в .NET Core</li> <li>▪ Продвинутое техники организации DAL</li> </ul>

Наименование тем	Виды учебных занятий	ак. час	Содержание
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Создание слоя данных для библиотеки книг</li> <li>▪ Реализация Unit of Work и транзакций</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: Изучение архитектуры приложений. Организация слоя для работы с БД</li> </ul>
Тема 3.7: Изучение архитектуры приложений. Организация бизнес-логики	Л	1	<ul style="list-style-type: none"> <li>▪ Зачем выделять BLL в отдельный слой</li> <li>▪ Принципы организации BLL</li> <li>▪ Service Layer Pattern</li> <li>▪ Domain Model Pattern</li> <li>▪ Валидация и обработка ошибок</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Создание BLL для банковской системы</li> <li>▪ Реализация системы скидок для интернет-магазина</li> </ul>
	СР	4	<ul style="list-style-type: none"> <li>▪ Задание: Изучение архитектуры приложений. Организация бизнес-логики</li> </ul>
Тема 3.8: Подходы к визуализации информации. Основы HTML и CSS	Л	1	<ul style="list-style-type: none"> <li>▪ Введение в веб-интерфейсы и визуализацию данных</li> <li>▪ Основы HTML. Структура веб-страницы</li> <li>▪ Основы CSS. Стилизация веб-интерфейсов</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Создание дашборда для CRM системы</li> <li>▪ Стилизация дашборда</li> </ul>
	СР	4	<ul style="list-style-type: none"> <li>▪ Задание: Подходы к визуализации информации. Основы HTML и CSS</li> </ul>
Тема 3.9: Работа с Blazor-проектом	Л	1	<ul style="list-style-type: none"> <li>▪ Введение в Blazor</li> <li>▪ Основы Razor компонентов</li> <li>▪ Маршрутизация и макеты</li> </ul>
	ПЗ	1	<ul style="list-style-type: none"> <li>▪ Создание шаблона CRM-приложения на Blazor</li> </ul>
	СР	1	<ul style="list-style-type: none"> <li>▪ Задание: Способы связи приложения с базами данных</li> </ul>
Тема 3.10: Изучение архитектуры приложений. Layout Pattern	Л	1	<ul style="list-style-type: none"> <li>▪ Проблемы, которые решает Layout Pattern</li> <li>▪ Принципы Layout Pattern</li> <li>▪ Реализация Layout Pattern в Blazor</li> <li>▪ Layout с поддержкой Breadcrumbs (хлебных крошек)</li> </ul>

Наименование тем	Виды учебных занятий	ак. час	Содержание
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Создание многоуровневого Layout для CRM системы</li> <li>▪ Создание Layout для аутентификации</li> </ul>
	СР	2	<ul style="list-style-type: none"> <li>▪ Задание: Способы связи приложения с базами данных</li> </ul>
Тема 3.11: Основы реализации автотестов. SeleniumWebDriver	Л	1	<ul style="list-style-type: none"> <li>▪ Какие проблемы решает автоматизированное тестирование</li> <li>▪ Типы автоматизированных тестов</li> <li>▪ Установка и настройка Selenium</li> <li>▪ Архитектура Selenium WebDriver</li> <li>▪ Основы написания автотестов</li> </ul>
	ПЗ	2	<ul style="list-style-type: none"> <li>▪ Тестирование формы входа</li> <li>▪ Тестирование CRUD операций с клиентами</li> </ul>
	СР	4	<ul style="list-style-type: none"> <li>▪ Задание: Способы связи приложения с базами данных</li> </ul>

### 1.7 Организационно-педагогические условия

Реализация программы осуществляется в полном соответствии с требованиями законодательства Российской Федерации в области образования, нормативными правовыми актами, регламентирующими данное направление деятельности.

#### 1.7.1 Требования к квалификации педагогических кадров

Реализация программы обеспечивается педагогическими работниками ООО «ЯРУС», а также приглашенными специалистами и действующими практиками IT-компаний, квалификация которых отвечает требованиям, указанным в квалификационных справочниках, и (или) профессиональных стандартах.

#### 1.7.2 Требования к материально-техническому обеспечению

При реализации программы используются дистанционные образовательные технологии и электронное обучение. ООО «ЯРУС» обеспечивает функционирование информационно-образовательной среды, включающей в себя электронные информационные ресурсы, электронные образовательные ресурсы, совокупность информационных технологий, телекоммуникационных технологий, соответствующих технологических средств и обеспечивающую освоение обучающимися программы полностью независимо от места нахождения обучающихся: каналы связи, компьютерное оборудование, периферийное оборудование, программное обеспечение.

Для использования программного обеспечения в рамках прохождения курса студенты должны обладать персональными компьютерами со следующими характеристиками:

Операционная система: Windows 10 или 11- 64-битная x86, 32-битная x86, или MacOS - 64-битная x86, или Linux - 64-битная x86, 64-битная Power8 / Power9;

Процессор 2.30 ГГц или быстрее, 2 ядра или больше;

Оперативная память 8+ Гб или больше;

Свободное место на жёстком диске 10 Гб и больше.

Компьютер с интернетом хорошего качества, камерой и микрофоном.

### 1.7.3 Требования к информационному и учебно-методическому обеспечению

Для реализации программы используются учебно-методическая документация, нормативные правовые акты, нормативная техническая документация, иная документация, учебная литература и иные издания, информационные ресурсы.

Таблица 6 – Учебно-методическая документация, нормативные правовые акты, нормативная техническая документация, иная документация, учебная литература и иные издания, информационные ресурсы.

<b>1. Нормативные правовые акты, иная документация</b>
1. Федеральный закон «Об образовании» от 29.12.2012 N 323-ФЗ
2. Указ Президента РФ от 02.07.2021 N 400 «О Стратегии национальной безопасности Российской Федерации»
3. Постановление Правительства Российской Федерации от 31.10.2002 № 787 «О порядке утверждения Единого тарифно-квалификационного справочника работ и профессий рабочих, Единого классификационного справочника должностей руководителей, специалистов и других служащих»
4. Приказ Минпросвещения России от 26.08.2020 N 438 «Об утверждении Порядка организации и осуществления образовательной деятельности по основным программам профессионального обучения»
5. Приказ Министерства образования и науки Российской Федерации от 12.09.2013 г. №1061 «Об утверждении перечней специальностей и направлений подготовки высшего образования»
6. Приказ Росстандарта «ОК 029-2014 (КДЕС Ред.2). Общероссийский классификатор

видов экономической деятельности» от 31.01.2014 г. №14ст
7. Приказ Минтруда России от 20.07.2022 N 424н «Об утверждении профессионального стандарта «Программист».
<b>2. Основная литература</b>
8. Рихтер Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд. – Издательский дом «Питер», 2025.
9. Стиллмен Э., Грин Д. Head First. Изучаем C#. 4-е изд. - Издательский дом «Питер», 2022.
10. Мартин Р. Чистый код: создание, анализ и рефакторинг. Библиотека программиста. - Издательский дом «Питер», 2025.
11. Мартин Р. Чистая архитектура. Искусство разработки программного обеспечения. - Издательский дом «Питер», 2025.
12. Бхаргава А. Грокаем алгоритмы. 2-е изд. - Издательский дом «Питер», 2025.
13. Хориков В. Принципы юнит-тестирования. - Издательский дом «Питер», 2025.
<b>3. Интернет-ресурсы</b>
14. C# documentation // URL: <a href="https://learn.microsoft.com/ru-ru/dotnet/csharp/">https://learn.microsoft.com/ru-ru/dotnet/csharp/</a>
15. Блог с подборкой статей практического характера // URL: <a href="https://code-maze.com/">https://code-maze.com/</a>
16. SQL documentation // URL: <a href="https://www.postgresql.org/docs/">https://www.postgresql.org/docs/</a>
17. Stackoverflow на русском // URL: <a href="https://ru.stackoverflow.com">https://ru.stackoverflow.com</a>

#### **1.7.4 Общие требования к организации учебного процесса**

Общие требования к организации учебного процесса определяются локальными нормативными актами ООО «ЯРУС».

#### **1.8 Формы аттестации**

Оценка качества освоения программы осуществляется в форме текущего контроля успеваемости, промежуточной аттестации по модулям и итоговой аттестации обучающихся по программе.

### 1.8.1 Текущий контроль успеваемости

Осуществляется в соответствии с учебным планом в форме оценки выполнения заданий текущего контроля (решения практических задач и выполнения практических работ (проектов)).

### 1.8.2 Итоговая аттестация

Освоение программы завершается итоговой аттестацией. Итоговая аттестация проводится на основе принципов объективности и независимости оценки качества подготовки обучающихся. Итоговая аттестация является обязательной для обучающихся.

Итоговая аттестация проводится в форме квалификационного экзамена. На квалификационном экзамене обучающиеся представляют защиту своего итогового практического проекта перед комиссией с элементами код-ревью, включающего в себя практическую и квалификационную работу и проверку теоретических знаний в пределах квалификационных требований к профессии «Программист».

К итоговой аттестации допускаются обучающиеся, не имеющие академической задолженности и в полном объеме выполнившие учебный план программы. Порядок прохождения квалификационного экзамена определяется локальными нормативными актами ООО «ЯРУС».

## 2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Оценочные материалы обеспечивают проверку достижения планируемых результатов обучения по программе и используются в процедуре текущего контроля успеваемости, промежуточной аттестации и итоговой аттестации.

### 2.1. Текущий контроль успеваемости

Реализуется в ходе проведения практических занятий в форме решения практических задач.

Таблица 7 – Соотнесение планируемых результатов с формами контроля и промежуточной аттестации

Название модуля	Планируемые результаты	Формы текущего контроля	Формы промежуточной аттестации
Модуль 1: Процедурное программирование	ПСК-1.1 ПСК-1.2	Решение практических задач	Выполнение практической работы (проекта)
Модуль 2: Объектно-Ориентированное программирование	ПСК-1.5 ПСК-2.1 ПСК-2.2 ПСК-2.3	Решение практических задач	Выполнение практической работы (проекта)
Модуль 3: Командная разработка коммерческих	ПСК-1.3 ПСК-1.4	Решение практических задач	Выполнение практической работы

приложений	ПСК-2.4 ПСК-2.5 ПСК-2.6		(проекта)
------------	-------------------------------	--	-----------

## 2.2. Промежуточная аттестация

Промежуточная аттестация реализуется в ходе проверки выполнения практической работы (проекта).

Таблица 8 – Содержание промежуточной аттестации

Название модуля	Планируемые результаты	Формы промежуточной аттестации	Тема проектного задания
Модуль 1: Процедурное программирование	ПСК-1.1 ПСК-1.2	Выполнение практической работы (проекта)	Реализация консольного калькулятора с историей операций
Модуль 2: Объектно-Ориентированное программирование	ПСК-1.5 ПСК-2.1 ПСК-2.2 ПСК-2.3	Выполнение практической работы (проекта)	Реализация Telegram-бота
Модуль 3: Командная разработка коммерческих приложений	ПСК-1.3 ПСК-1.4 ПСК-2.4 ПСК-2.5 ПСК-2.6	Выполнение практической работы (проекта)	Проектирование и реализация CRM-проекта

**Порядок проведения:** в практической работе (проекте) проверяется правильность кодирования по поставленной задаче в рамках темы проекта.

Практическая работа (проект) считается выполненной и ставится оценка «зачтено» по промежуточной аттестации в случае наличия в практической работе (проекте) прописанных работающих кодов по поставленной задаче.

## 2.3. Итоговая аттестация

Итоговая аттестация в форме квалификационного экзамена имеет целью определить сформированность спланированных к освоению профессиональных компетенций, связанных с коммерческой разработкой программных приложений.

Квалификационный экзамен представляет собой защиту итоговой практической работы (проекта) по разработке программного продукта перед комиссией с элементами код-ревью.

Таблица 9 - Спецификация заданий для проверки умений и навыков

Предмет оценки (умение, навык)	Критерии оценки	Шкала оценки	Тип задания
1	2	3	4
Работа с типами и операторами	Созданы переменные разных	2 балла – прописан	Кодирование

Предмет оценки (умение, навык)	Критерии оценки	Шкала оценки	Тип задания
1	2	3	4
	типов, выполнены арифметические операции, используется ввод/вывод	работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	
Использование условных конструкций	Логика корректно реализована через if/else или switch, учтены граничные условия	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Применение циклов	Решение использует соответствующие циклы, корректно обработаны условия выхода	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Создание методов	Созданы методы с правильной сигатурой, возвращаемые значения соответствуют заданию	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Работа с Git	Создан репозиторий, коммиты отражают логические этапы работы, есть осмысленное описание	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Работа с массивами и коллекциями	Создан и заполнен массив, реализованы операции поиска, фильтрации	2 балла – прописан работающий код; 1 балл – код прописан, но работает не	Кодирование

Предмет оценки (умение, навык)	Критерии оценки	Шкала оценки	Тип задания
1	2	3	4
		корректно, 0 баллов – код отсутствует	
Написание unit- тестов	Созданы тестовые случаи (позитивные и негативные), покрывающие основные сценарии, высокое покрытие кода	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Проектирование класса	Создан класс с полями, свойствами, методами, конструкторами; реализована инкапсуляция	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Реализация наследования и полиморфизма	Построена корректная иерархия классов, используются виртуальные методы и переопределение	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Работа с интерфейсами и абстрактными классами	Созданы и реализованы интерфейсы, использованы абстрактные классы, продемонстрирован полиморфизм	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Создание 2- слойной архитектуры	Четкое разделение слоев, логичная организация кода	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Проектирование	Создана	2 балла –	Кодирование

Предмет оценки (умение, навык)	Критерии оценки	Шкала оценки	Тип задания
1	2	3	4
БД	нормализованная схема данных	прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	
Написание SQL-запросов	Реализованы CRUD-операции, сложные запросы с JOIN и агрегацией	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Создание DAL с Dapper	Реализован слой доступа к данным корректное использование Dapper, работа с параметрами	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Реализация BLL	Создан слой бизнес-логики, обеспечено покрытие тестами	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование
Создание UI на Blazor	Реализован функциональный интерфейс с компонентами Blazor, привязка данных к BLL	2 балла – прописан работающий код; 1 балл – код прописан, но работает не корректно, 0 баллов – код отсутствует	Кодирование

**Максимальное количество баллов – 32.**

**Порядок проведения:** защита итоговой практической работы (проекта) проводится с личного компьютера, в ходе кода-ревью оценивается правильность прописанных кодов.

Защита итоговой практической работы (проекта) оценивается зачтено/ не зачтено.

Для получения оценки «зачтено» необходимо в итоговой практической работе (проекте) прописать работающие коды по всем 16 предметам оценки и набрать не менее 22 баллов.

В случае получения оценки «зачтено» квалификационный экзамен считается сданным.