

Документ подписан простой электронной подписью

Информация о документе:

ФИО: Хоружий Людмила Ивановна

Должность: Директор института экономики и управления АПК

Дата подписания: 28.08.2025 14:44:57

Уникальный программный ключ:

1e90b132d9b04dceb7585160b015dddf2cb1e6a9



МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ – МСХА имени К.А. ТИМИРЯЗЕВА»

(ФГБОУ ВО РГАУ - МСХА имени К.А. Тимирязева)

Институт экономики и управления АПК
Кафедра статистики и кибернетики

УТВЕРЖДАЮ:

Директор института
экономики и управления АПК

Л.И. Хоружий



«28» августа 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.31 Программная инженерия

для подготовки бакалавров

ФГОС ВО

Направление 09.03.02 Информационные системы и технологии

Направленность: Компьютерные науки и технологии искусственного интеллекта

Курс 4, 5

Семестр 2, 3

Форма обучения: заочная

Год начала подготовки: 2025

Москва, 2025

Разработчик (и):
Демичев В.В., к.э.н., доцент _____


«26» августа 2025г.

Титов А.Д., ассистент


«26» августа 2025г.

Рецензент: Чепурина Е.Л., к.э.н., доцент


«26» августа 2025г.

Программа составлена в соответствии с требованиями ФГОС ВО, профессионального стандарта и учебного плана по направлению подготовки 09.03.02 «Информационные системы и технологии» и учебного плана по данному направлению.

Программа обсуждена на заседании кафедры статистики и кибернетики протокол № 11 от «26» августа 2025 г.

И.О. зав. кафедрой статистики и кибернетики
Уколова А.В. к.э.н., доцент


«26» августа 2025г.

Согласовано:

Председатель учебно-методической
комиссии института экономики и управления АПК
Гупалова Т.Н., к. э. н, доцент


протокол № 1 «28» августа 2025г.

И.О. зав. кафедрой статистики и кибернетики
Уколова А.В. к.э.н., доцент


«26» августа 2025г.

Заведующий отделом комплектования ЦНБ


(подпись)

СОДЕРЖАНИЕ

АННОТАЦИЯ.....	4
1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ	5
2. МЕСТО ДИСЦИПЛИНЫ В УЧЕБНОМ ПРОЦЕССЕ	5
3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	5
4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	5
4.1 РАСПРЕДЕЛЕНИЕ ТРУДОЁМКОСТИ ДИСЦИПЛИНЫ ПО ВИДАМ РАБОТ	5
ПО СЕМЕСТРАМ	5
4.2 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	12
4.3 ЛЕКЦИИ/ ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	14
5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ.....	16
6. ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ	17
6.1. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ.....	17
6.2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ КОНТРОЛЯ УСПЕВАЕМОСТИ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ	20
7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	21
7.1 ОСНОВНАЯ ЛИТЕРАТУРА.....	21
7.2 ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА	21
8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО- ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)	22
9. ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ (ПРИ НЕОБХОДИМОСТИ)	22
10. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ).....	22
11. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ	24
12. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПРЕПОДАВАТЕЛЯМ ПО ОРГАНИЗАЦИИ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ	25

Аннотация

рабочей программы учебной дисциплины Б1.О.31 «Программная инженерия» для подготовки бакалавров по направлению 09.03.02 «Информационные системы и технологии» направленности «Компьютерные науки и технологии искусственного интеллекта»

Цель освоения дисциплины:

Целью освоения дисциплины «Программная инженерия» является формирование у студентов теоретических и практических навыков по изучению и использованию современных технологий разработки программного обеспечения в соответствии с международными стандартами обучения программной инженерии. Основными задачами изучения дисциплины являются формирование у студентов умений и навыков по проблемам оценки требований, проектирования, разработки, качества, повышения надежности и документирования программного обеспечения, а также по вопросам управления коллективной разработкой программного обеспечения.

Место дисциплины в учебном плане: Дисциплина включена в часть, формируемую участниками образовательных отношений учебного плана по направлению подготовки 09.03.02 "Информационные системы и технологии".

Требования к результатам освоения дисциплины: в результате освоения дисциплины формируются следующие компетенции (индикаторы):

Краткое содержание дисциплины: Основные понятия, SWEBOK (Software Engineering Body of Knowledge). Жизненный цикл программного обеспечения: классические и гибкие модели (Waterfall, Spiral, Agile, DevOps, MLOps). Сбор, анализ и документирование требований: функциональные и нефункциональные требования, спецификация требований (SRS), стандарты IEEE 830 и ГОСТ 19. Моделирование и проектирование ПО с использованием UML: диаграммы прецедентов, классов, последовательности, компонентов. Объектно-ориентированный анализ и проектирование. Архитектурные стили и паттерны: монолит, микросервисы, MVC. Принципы модульности: cohesion (внутренняя связность) и coupling (внешняя связанность). Гибкие методологии разработки. Инструменты управления проектами. Системы контроля версий. Качество кода и рефакторинг. Тестирование ПО: уровни, виды, подходы (TDD, BDD), инструменты. Особенности тестирования ИИ-систем. Техническая документация: виды, требования ГОСТ 34 и ГОСТ 19, оформление РП и РУ. Управление качеством процессов разработки: модели зрелости (CMMI), стандарт ISO 9001. Отечественные программные средства и стандарты в контексте импортозамещения.

Общая трудоемкость дисциплины составляет: 3 зачетные единицы (108 часов).

Промежуточный контроль: экзамен.

1. Цель освоения дисциплины

Целью освоения дисциплины «Программная инженерия» является формирование у студентов теоретических и практических навыков по изучению и использованию современных технологий разработки программного обеспечения в соответствии с международными стандартами обучения программной инженерии. Основными задачами изучения дисциплины являются формирование у студентов умений и навыков по проблемам оценки требований, проектирования, разработки, качества, повышения надежности и документирования программного обеспечения, а также по вопросам управления коллективной разработкой программного обеспечения.

2. Место дисциплины в учебном процессе

Дисциплина «Программная инженерия» включена в часть, формируемую участниками образовательных отношений учебного плана. Дисциплина «Программная инженерия» реализуется в соответствии с требованиями ФГОС ВО и Учебного плана по направлению 09.03.02 «Информационные системы и технологии». Предшествующими курсами, на которых базируется дисциплина, являются: «Системное программирование», «Интернет-программирование», «Высокоуровневое программирование», «Операционные системы». Последующие дисциплины: «Методы искусственного интеллекта», «Администрирование информационных систем», «ERP-системы в управлении бизнесом».

Дисциплина «Программная инженерия» может быть использована при написании выпускной квалификационной работы.

Рабочая программа дисциплины «Программная инженерия» для инвалидов и лиц ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психологического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся компетенций, представленных в таблице 1.

4. Структура и содержание дисциплины

4.1 Распределение трудоёмкости дисциплины по видам работ по семестрам

Общая трудоёмкость дисциплины составляет 3 зач.ед. (108 часов), их распределение по видам работ семестрам представлено в таблице 2.

Требования к результатам освоения учебной дисциплины

№ п/п	Код компетенции	Содержание компетенции (или её части)	Индикаторы компетенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
1.	ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Знать: современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности	современные информационные технологии и программные средства в сфере профессиональной деятельности		
			ОПК-2.2 Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности		выбирать современные информационные технологии и программные средства при решении задач профессиональной деятельности	

№ п/п	Код компетенции	Содержание компетенции (или её части)	Индикаторы компетенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
			ОПК-2.3 Иметь навыки: применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности			навыками применения современных информационных технологий и программных средств при решении задач профессиональной деятельности
2.	ОПК-4	Способен участвовать в разработке технической документации, связанной с профессиональной деятельностью с использованием стандартов, норм и правил	ОПК-4.1 Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы	основные методы и законы моделирования информационных процессов и систем, принципы моделирования, классификацию способов представления моделей систем, достоинства и недостатки различных способов представления моделей систем, алгоритмы		

№ п/п	Код компетенции	Содержание компетенции (или её части)	Индикаторы компетенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
				фиксации и обработки результатов моделирования систем, способы планирования машинных экспериментов с моделями.		
			ОПК-4.2 Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы		использовать технологии моделирования, представлять модель в математическом и алгоритмическом виде, оценивать качество модели, показывать теоретические основания модели, проводить статистическое моделирование систем, моделировать процессы, протекающие в информационных системах и сетях.	
			ОПК-4.3 Иметь навыки: составле-			навыками построения имитационных

№ п/п	Код компетенции	Содержание компетенции (или её части)	Индикаторы компетенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
			ния технической документации на различных этапах жизненного цикла информационной системы			моделей информационных процессов, методологией получения концептуальных моделей систем, навыками построения моделирующих алгоритмов
3.	ОПК-8	Способен применять математические модели, методы и средства проектирования информационных и автоматизированных систем	ОПК-8.1 Знать: методологию и основные методы математического моделирования, классификацию и условия применения моделей, основные методы и средства проектирования информационных и автоматизированных систем, инструментальные средства моделирования и проек-	основы теории математического моделирования; классификацию, свойства и области применения математических моделей; принципы проектирования информационных и автоматизированных систем; методы системного анализа и формального описания требований; современные инструментальные		

№ п/п	Код компе- тенции	Содержание компетенции (или её ча- сти)	Индикаторы ком- петенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
			тирования	средства моделиро- вания и проектиро- вания		
			ОПК-8.2 Уметь: применять на практике матема- тические модели, методы и средства проектирования и автоматизации систем на практике		формализовать при- кладные задачи в ви- де математических моделей; выбирать адекватные методы моделирования и проектирования в за- висимости от типа системы и постав- ленной цели; приме- нять инструменталь- ные средства для по- строения, анализа и валидации моделей; разрабатывать архи- тектуру и структуру информационных и автоматизированных систем с учётом тре- бований масштаби- руемости, надёжно- сти и эффективно- сти.	

№ п/п	Код компе- тенции	Содержание компетенции (или её ча- сти)	Индикаторы ком- петенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
			ОПК-8.3 Иметь навыки: моделиро- вания и проекти- рования информа- ционных и автома- тизированных систем			навыками построения и верификации математических моделей реальных процессов и систем; методиками проектирования компонентов информационных и автоматизированных систем; практическими приёмами работы с программными средствами моделирования и проектирования; навыками документирования проектных решений и представления результатов моделирования заинтересованным сторонам

Таблица 2

Распределение трудоёмкости дисциплины по видам работ по семестрам

Вид учебной работы	Трудоёмкость		
	Всего, час.	в т.ч. по семестрам	
		4	5
Общая трудоёмкость дисциплины по учебному плану	108	36	72
1. Контактная работа:	14,4	2	12,4
Аудиторная работа	14,4	2	12,4
<i>в том числе:</i>			
<i>лекции (Л)</i>	6	2	4
<i>практические занятия (ПЗ)</i>	8		8
<i>контактная работа на промежуточном контроле (КРА)</i>	0,4		0,4
2. Самостоятельная работа (СРС)	85	34	51
<i>самостоятельное изучение разделов, самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к лабораторным и практическим занятиям, коллоквиумам и т.д.)</i>	85	34	51
<i>Подготовка к экзамену (контроль)</i>	8,6		8,6
Вид промежуточного контроля:	Экзамен		

4.2 Содержание дисциплины

Таблица 3

Тематический план учебной дисциплины

Наименование разделов и тем дисциплин (укрупнённо)	Всего	Аудиторная работа			Внеаудиторная работа СР
		Л	ПЗ	ПКР	
4 семестр					
Тема 1. Введение в программную инженерию	36	2			34
Всего за 4 семестр	36	2			34
5 семестр					
Тема 2. Жизненный цикл программного обеспечения. Сбор и анализ требований. Документирование требований и технических спецификаций. Моделирование ПО с использованием UML	27	2	2		23
Тема 3. Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ	44,6	2	6		36,6
контактная работа на промежуточном контроле (КРА)	0,4			0,4	

Наименование разделов и тем дисциплин (укрупнённо)	Всего	Аудиторная работа			Внеаудиторная работа СР
		Л	ПЗ	ПКР	
Всего за 5 семестр	72	4	8	0,4	59,6
Итого по дисциплине	108	6	8	0,4	93,6

Тема 1 Введение в программную инженерию

Определение программной инженерии: цели, задачи. Программное обеспечение (программный продукт). SWEBOK (Software Engineering Body of Knowledge) как международный стандарт знаний — структура, ключевые области (требования, проектирование, тестирование, управление и др.). Эволюция методологий разработки (классические, итеративные, гибкие, современные). Необходимость версионирования моделей и данных. Обзор отечественных ИТ-решений: платформы, CASE-средства, системы контроля версий, облачные решения.

Тема 2 Жизненный цикл программного обеспечения. Сбор и анализ требований. Документирование требований и технических спецификаций. Моделирование ПО с использованием UML

Понятие жизненного цикла ПО, этапы. Сравнительный анализ моделей ЖЦ. Влияние выбора модели ЖЦ на: состав команды, инструменты, формат документации, подход к тестированию.

Источники требований. Методы сбора требований. Классификация требований с точки зрения системы. Требования к ИИ-системам. Анализ противоречий и приоритизация требований. Работа с заинтересованными сторонами.

Структура документа SRS (Software Requirements Specification): введение, общее описание, функциональные/нефункциональные требования, приложения. Стандарты IEEE 830, ГОСТ 19.101–19.701, ISO/IEC/IEEE 29148. Требования к оформлению. Инструменты документирования.

Назначение и преимущества визуального моделирования. Основные типы диаграмм UML. Принципы объектно-ориентированного анализа и проектирования. Инструменты моделирования.

Тема 3 Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ

Принципы проектирования. Архитектурные стили и паттерны. Применение математических моделей при проектировании: модели очередей, надёжности, графовые модели.

Принципы Agile (Agile Manifesto). Scrum: роли, артефакты. Инструменты управления. Короткие эксперименты, метрики качества моделей как критерии готовности.

Принципы чистого кода (Clean Code): именование, функции, комментарии, форматирование. Стандарты кодирования. Системы контроля версий. Инструменты анализа качества. Code review: практики, польза, автоматизация.

Уровни и подходы тестирования. Инструменты тестирования. Метрики качества (accuracy, precision, recall, F1). Проверка на смещённых данных. A/B-тестирование моделей. Автоматизация тестирования.

Виды документации по этапам ЖЦ: техническое задание (ТЗ) – ГОСТ 19.201, архитектурное описание (АТ) – ГОСТ 19.402, руководство программиста (РП) – ГОСТ 19.504, руководство пользователя (РУ) – ГОСТ 19.505. Требования ГОСТ к автоматизированным системам. Современные подходы. Работа с документацией в команде: версионирование, рецензирование.

Понятие качества ПО: соответствие требованиям, надёжность, удобство, эффективность. Модели зрелости: CMMI (Capability Maturity Model Integration), ISO 9001. Внутренние метрики. Аудит процессов разработки: цели, этапы, отчётность.

4.3 Лекции/ практические занятия

Таблица 4

Содержание лекций/практических занятий и контрольные мероприятия

№ п/п	Название раздела, темы	№ и название лекций/практических занятий	Формируемые компетенции	Вид контрольного мероприятия	Кол-во Часов
4 семестр					
1.	Тема 1. Введение в программную инженерию	Лекция №1. Введение в программную инженерию	ОПК-2.1, ОПК-8.1		2
5 семестр					
2.	Тема 2. Жизненный цикл ПО. Сбор и анализ требований. Документирование требований. Моделирование ПО с использованием UML	Лекция №2. Сбор и анализ требований к ПО. Стандарты и оформление технической документации. Моделирование систем с использованием UML	ОПК-4.1, ОПК-2.3, ОПК-8.1, ОПК-8.2, ОПК-8.3		2
		Практическая работа № 1. Сравнение моделей жизненного цикла ПО. Выявление и формулирование требований для проекта. Разработка спецификации требований (SRS) по ГОСТ 19.201.	ОПК-2.2, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-8.1, ОПК-8.2	защита практической работы, круглый стол	2
3.	Тема 3. Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ	Практическая работа № 2. Построение диаграмм UML. Проектирование архитектуры ИС	ОПК-8.3, ОПК-2.3, ОПК-8.2	защита практической работы	2

№ п/п	Название раздела, темы	№ и название лекций/практических занятий	Формируемые компетенции	Вид контрольного мероприятия	Кол-во Часов
		Практическая работа № 3. Организация работы команды в Agile-процессах: планирование спринта и ведение бэклога в международных и отечественных системах управления проектами. Работа с Git и отечественными системами контроля версий, рефакторинг, настройка	ОПК-2.3, ОПК-8.2	защита практической работы	2
		Лекция №3. Архитектура и проектирование информационных систем. Гибкие методологии разработки. Кодирование, контроль версий и качество кода	ОПК-2.2, ОПК-2.3, ОПК-8.1, ОПК-8.2		2
		Практическая работа № 4. Тестирование ПО: unit-тесты (JUnit, pytest), TDD. Оформление технической документации по ГОСТ (Руководство программиста, Руководство пользователя)	ОПК-2.3, ОПК-8.3, ОПК-4.2, ОПК-4.3	защита практической работы, дискуссия	2

4.4 Перечень вопросов для самостоятельного изучения дисциплины

Таблица 5

Перечень вопросов для самостоятельного изучения дисциплины

№ п/п	Название раздела, темы	Перечень рассматриваемых вопросов для самостоятельного изучения
4 семестр		
1.	Тема 1. Введение в программную инженерию	Роль программной инженерии в разработке ИИ-систем. Отличие SWEBOOK от других стандартов. Отечественные инициативы в области импортозамещения ПО. ОПК-2.1, ОПК-8.1
5 семестр		
2.	Тема 2. Жизненный цикл ПО. Сбор и анализ требований. Документирование требований. Моделирование ПО с использованием UML	Сравните модели жизненного цикла ПО по критериям: гибкость, предсказуемость, применимость в ИИ-проектах. Особенности жизненного цикла MLOps по сравнению с традиционным DevOps. Водопадная модель в госсекторе. Способы выявления нефункциональных требований в проектах с ИИ. Этические и юридические ограничения при формулировании требований к ИИ-системам (с учётом ФЗ-152 «Об обработке персональных данных», рекомендаций UNESCO). Приоритизация требований по методу MoSCoW. Отличия оформления технической документации по ГОСТ от международных практик (IEEE, ISO). Отечественные инструменты для ведения требований в соответствии с ЕСПД. Какие диаграммы UML наиболее полезны на этапе анализа требований, а какие – на этапе проектирования. Отечественные CASE-средства, поддерживающие нотацию UML и совместимые с российскими ОС. Построение диаграммы последовательности для сценария «Пользователь получает

№ п/п	Название раздела, темы	Перечень рассматриваемых вопросов для самостоятельного изучения
		ИИ-рекомендацию». ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-8.1, ОПК-8.2
3.	Тема 3. Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ	Выбор архитектурного паттерна (микросервисы, монолит) для системы с ИИ-компонентом. Как математические модели (например, теория массового обслуживания) могут использоваться при проектировании ИС. Адаптация Scrum для проекта с нестабильными требованиями. Функции управления задачами и бэклог в «Битрикс24» и «Мегаплане» по сравнению с Jira. Риски возникают при применении Agile в госзаказе. Принципы «чистого кода» для командной разработки. Отечественные системы контроля версий и их отличие от Git. Настройка автоматической проверки кода. Метрики качества для оценки ИИ-модели. Отличие этапа тестирования традиционного ПО от ИИ-систем. Отечественные инструменты автоматизированного тестирования. Требования к Руководству программиста и Руководству пользователя по ГОСТ 19.504 и 19.505. Как автоматизировать генерацию документации из исходного кода (на примере Sphinx). Уровни зрелости, определяемые моделью CMMI. Какой уровень типичен для российских ИТ-компаний. Стандарт ISO 9001 при разработке ПО в РФ. Меры по повышению зрелости процессов для стартапа в области ИИ., ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-8.1, ОПК-8.2, ОПК-8.3

5. Образовательные технологии

Таблица 6

Применение активных и интерактивных образовательных технологий

№ п/п	Тема и форма занятия	Наименование используемых активных и интерактивных образовательных технологий
1.	Тема 2. Жизненный цикл ПО. Сбор и анализ требований. Документирование требований. Моделирование ПО с использованием UML	ПЗ Круглый стол
2.	Тема 3. Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ	ПЗ Дискуссия

6. Текущий контроль успеваемости и промежуточная аттестация по итогам освоения дисциплины

6.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности

1) Пример тестовых заданий

Тема 3. Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ

Какие принципы относятся к Agile-манифесту?

- a) Процессы и инструменты важнее людей
- b) Работающий продукт важнее исчерпывающей документации
- c) Следование плану важнее реагирования на изменения
- d) Сотрудничество с заказчиком важнее согласования условий контракта

Что такое Product Backlog в Scrum?

- a) Список задач на текущий спринт
- b) Упорядоченный список всего, что может понадобиться в продукте
- c) Отчёт о завершённых задачах
- d) План релиза

Какие из перечисленных инструментов являются отечественными системами управления проектами?

- a) Jira
- b) «Битрикс24»
- c) Trello
- d) «Мегаплан»

Какая команда Git создаёт новую ветку?

- a) `git checkout -b имя_ветки`
- b) `git branch имя_ветки`
- c) `git merge имя_ветки`
- d) `git clone`

Что проверяет статический анализатор кода SonarQube?

- a) Покрытие тестами
- b) Наличие уязвимостей безопасности
- c) Стиль кодирования
- d) Все перечисленные варианты

Какой подход предполагает написание тестов до реализации кода?

- a) BDD
- b) TDD

- c) CI/CD
- d) MLOps

Какие метрики качества характерны для оценки ИИ-моделей?

- a) Accuracy
- b) Покрытие требований
- c) F1-score
- d) Время отклика API

Какие из перечисленных фреймворков используются для unit-тестирования в Python?

- a) JUnit
- b) pytest
- c) TestNG
- d) unittest

Какие принципы «чистого кода» (Clean Code) вы знаете?

- a) Функции должны быть короткими и выполнять одну задачу
- b) Имена переменных должны быть понятными
- c) Комментарии заменяют плохой код
- d) Избегать дублирования кода

2) Примерный перечень дискуссионных тем для круглого стола

Тема 2. Жизненный цикл ПО. Сбор и анализ требований. Документирование требований. Моделирование ПО с использованием UML

1. «UML: устаревший инструмент или незаменимый стандарт?»
Роль визуального моделирования в эпоху быстрого прототипирования.
2. «Отечественные CASE-средства: реальность и перспективы замены StarUML и Enterprise Architect»
Сравнение функциональности, поддержка ГОСТ, интеграция с российскими ОС.
3. «Как проектировать ИИ-системы без архитектурного хаоса?»
Проблемы монолитов, pipeline-подход, управление версиями моделей и данных.

3) Примеры практических работ

Тема 2. Жизненный цикл ПО. Сбор и анализ требований. Документирование требований. Моделирование ПО с использованием UML

1. Диаграмма вариантов использования.
2. Диаграмма классов.
3. Диаграммы коопераций и последовательностей.

4. Диаграмма состояний.
5. Диаграмма деятельности.
6. Диаграмма компонентов.
7. Диаграмма развертывания.

Тема 3. Архитектура и проектирование ИС. Гибкие методологии и командная работа. Кодирование и качество кода. Тестирование ПО. Техническая документация на всех этапах ЖЦ

1. Анализ и выбор архитектурного паттерна.
2. Проектирование системы с применением математических моделей.
3. Сравнение архитектур: монолит и микросервисы в условиях госзаказа.
4. Разработка архитектурного описания по ГОСТ 19.402.

Вопросы к экзамену

1. Что такое программная инженерия?
2. Опишите структуру и назначение SWEBOOK.
3. В чём суть Agile-манифеста? Назовите его 4 ценности.
4. Какие отечественные ИТ-решения применяются в госсекторе? Приведите примеры.
5. Чем отличаются функциональные и нефункциональные требования?
6. Какие разделы обязательны в SRS по ГОСТ 19.201?
7. Какие методы выявления требований наиболее эффективны при работе с неопытными заказчиками?
8. Требования ФЗ-152 «Об обработке персональных данных» и рекомендации UNESCO?
9. Какие виды ограничений необходимо учитывать при сборе требований?
10. Как приоритизировать требования по методу MoSCoW?
11. Какие инструменты поддерживают ведение документации по ЕСПД?
12. Для чего используется диаграмма последовательности в UML?
13. Что отображает диаграмма последовательности? Приведите пример.
14. В чём преимущества микросервисной архитектуры перед монолитной?
15. Архитектурные паттерны
16. Какие отечественные CASE-средства вы знаете?
17. Как обеспечить отказоустойчивость?
18. Назовите роли и артефакты в Scrum.
19. Чем Kanban отличается от Scrum?
20. Какие артефакты используются в Scrum-процессе?
21. Какие функции управления задачами есть в «Битрикс24» и «Мегаплане»?
22. Какие команды Git используются для работы с ветками?
23. Что проверяет SonarQube?
24. В чём отличие unit-тестов от интеграционных?
25. Какие метрики качества используются для оценки моделей?
26. Почему тестирование ИИ-систем сложнее, чем традиционного ПО?
27. Какие разделы обязательны в Руководстве программиста по ГОСТ 19.504?

28. Какие требования предъявляются к Руководству пользователя по ГОСТ 19.505?
29. Как можно автоматизировать генерацию документации из исходного кода?
30. Какие метрики эффективности используются для оценки работы Agile-команды?
31. Что такое СММІ? Назовите уровни зрелости.
32. Как стандарт ISO 9001 применяется в разработке ПО?
33. Какие меры можно предложить для повышения зрелости процессов в российской ИТ-компании?
34. Какие модели ЖЦ применяются в проектах с нестабильными требованиями?
35. Возможно ли применение Agile в госзаказе? Обоснуйте.
36. Принципы «чистого кода».
37. Какие отечественные системы контроля версий существуют?

6.2. Описание показателей и критериев контроля успеваемости, описание шкал оценивания

Для оценки знаний, умений, навыков и формирования компетенции по дисциплине может применяться **балльно-рейтинговая/традиционная** система контроля и оценки успеваемости студентов.

В основу балльно-рейтинговой системы (БРС) положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется в ходе текущего, промежуточного контроля и промежуточной аттестации знаний.

Таблица 7

Баллы	Балльная оценка текущей успеваемости			
За круглый стол	2	3	4	5
За дискуссию	2	3	4	5
За тестирование	2	6	7	8
За практическую работу	0	1	2	3
За экзамен	2	3	4	5
Оценка	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично

Критерии оценивания результатов обучения

Таблица 8

Оценка	Шкала оценивания	Критерии оценивания
Высокий уровень «5» (отлично)	85-100	оценку «отлично» заслуживает студент, освоивший знания, умения, компетенции и теоретический материал без пробелов; выполнивший все задания, предусмотренные учебным планом на высоком качественном уровне; практические навыки профессионального применения освоенных знаний сформированы. Компетенции, закреплённые за дисциплиной, сформированы на уровне – высокий.

Средний уровень «4» (хорошо)	70-84	оценку «хорошо» заслуживает студент, практически полностью освоивший знания, умения, компетенции и теоретический материал, учебные задания не оценены максимальным числом баллов, в основном сформировал практические навыки. Компетенции, закреплённые за дисциплиной, сформированы на уровне – хороший (средний).
Пороговый уровень «3» (удовлетворительно)	61-69	оценку «удовлетворительно» заслуживает студент, частично с пробелами освоивший знания, умения, компетенции и теоретический материал, многие учебные задания либо не выполнил, либо они оценены числом баллов близким к минимальному, некоторые практические навыки не сформированы. Компетенции, закреплённые за дисциплиной, сформированы на уровне – достаточный.
Минимальный уровень «2» (неудовлетворительно)	0-60	оценку «неудовлетворительно» заслуживает студент, не освоивший знания, умения, компетенции и теоретический материал, учебные задания не выполнил, практические навыки не сформированы. Компетенции, закреплённые за дисциплиной, не сформированы.

7. Учебно-методическое и информационное обеспечение дисциплины

7.1 Основная литература

1. Казарин О., Шубинский И. Надежность и безопасность программного обеспечения 2-е изд. Учебник для вузов. – Litres, 2025.
2. Самочадин А. В., Самочадина Т. Н. Оценка трудоемкости разработки программного обеспечения: учебное пособие. – 2023.
3. Гома Х. UML. Проектирование систем реального времени, распределенных и параллельных приложений. – Litres, 2022.
4. Проскураков А. Качество и тестирование программного обеспечения. Метрология программного обеспечения. – Litres, 2022.
5. Маран, М.М. Программная инженерия: учебное пособие / М.М. Маран. – Санкт-Петербург: Лань, 2018. – 196 с.

7.2 Дополнительная литература

1. Вейцман, В.М. Проектирование информационных систем: учебное пособие / В.М. Вейцман. — Санкт-Петербург: Лань, 2019. — 316 с. — ISBN 978-5-8114- 3713-9. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/122172> — Загл. с экрана.
2. Водяхо, А.И. Архитектурные решения информационных систем: учебник / А.И. Водяхо, Л.С. Выговский, В.А. Дубенецкий, В.В. Цехановский. — 2-е изд., перераб. — Санкт-Петербург: Лань, 2017. — 356 с. — ISBN 978-5-8114- 2556-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/96850> — Загл. с экрана.
3. Гвоздева, Т.В. Проектирование информационных систем. Стандартизация: учебное пособие / Т.В. Гвоздева, Б.А. Баллод. — Санкт-Петербург: Лань, 2019. — 252 с. — ISBN 978-5-8114-3517-3. — Текст: электронный //

Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/115515> — Загл. с экрана.

4. Зубкова, Т.М. Технология разработки программного обеспечения: учебное пособие / Т.М. Зубкова. — Санкт-Петербург: Лань, 2019. — 324 с. — ISBN 978-5-8114-3842-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/122176> — Загл. с экрана.
5. Остроух, А.В. Проектирование информационных систем: монография / А.В. Остроух, Н.Е. Суркова. — Санкт-Петербург: Лань, 2019. — 164 с. — ISBN 978-5-8114-3404-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/118650> — Загл. с экрана.

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

1. <https://gostinform.ru/> Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. (открытый доступ)
2. <http://opdo.timacad.ru> Система дистанционного обучения РГАУ МСХА им. К.А. Тимирязева (открытый доступ)
3. <http://msdn.microsoft.com/> (открытый доступ)
4. <http://www.rsdn.ru/> Российский журнал для программистов (открытый доступ)

9. Перечень программного обеспечения и информационных справочных систем

Таблица 9

Перечень программного обеспечения

№ п/п	Наименование раздела учебной дисциплины (модуля)	Наименование программы	Тип программы	Автор	Год разработки
1	Тема 2-3	Microsoft Office	Демонстрирующая	Microsoft	2017
2	Тема 2-3	StarUML	CASE-система / инструмент моделирования UML	MKLab / StarUML Team	2025
3	Тема 3	GitHub	Платформа для хостинга репозиторий	GitHub (Microsoft)	2025
4	Тема 3	Sphinx	Генератор документации	Georg Brandl и сообщество	2025

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

**Сведения об обеспеченности специализированными аудиториями,
кабинетами, лабораториями**

Наименование специальных помещений и помещений для самостоятельной работы (№ учебного корпуса, № аудитории)	Оснащенность специальных помещений и помещений для самостоятельной работы
1	2
<i>учебная аудитория для проведения занятий лекционного типа, групповых и индивидуальных консультаций (2й учебный корпус, 102 ауд.)</i>	<ol style="list-style-type: none"> 1. Компьютер – 29 шт.; 2. Стенд «Сергеев Сергей Степанович 1910-1999» (Инв. №591013/25) – 1 шт.; 3. Огнетушитель порошковый (Инв. №559527) – 1 шт.; 4. Подвесное крепление к огнетушителю (Инв. № 559528) – 1 шт.; 5. Жалюзи (Инв. №1107-221225, Инв. №1107-221225) – 2шт.; 6. Стул – 29 шт.; 7. Стол компьютерный – 28 шт.; 8. Стол для преподавателя – 1 шт.; 9. Доска маркерная (Инв. № 558762/5) – 1 шт.; 10. Трибуна напольная (без инв. №) – 1 шт.
<i>учебная аудитория для текущего контроля и промежуточной аттестации (2й учебный корпус, 106 ауд.)</i>	<ol style="list-style-type: none"> 1. Рабочая станция FORSITE TH1516G512G, Российская Федерация A4Tech Fstyler F1512 – 16 шт.; 2. Стол наборный (Инв. №410136000010828) – 1 шт. 3. Стол компьютерный (Инв. № 410136000010813-410136000010827) – 15 шт.; 4. Стул (Инв. № 410136000010829-410136000010853) – 25 шт.; 5. Интерактивная панель (Инв. № 410124000603715) – 1 шт.
<i>учебная аудитория для групповых и индивидуальных консультаций, учебная аудитория для текущего контроля и промежуточной аттестации</i>	<ol style="list-style-type: none"> 1. Компьютер – 16 шт. 2. Телевизор – 1 шт. 3. Стол для преподавателя – 1 шт. 4. Стол компьютерный – 16 шт. 5. Стул офисный – 17 шт. 6. Компьютер: PRO-3159209 Intel Core i5-10400 2900МГц, Intel B460, 16Гб DDR4, Intel UHD Graphics 630 (встроенная), SSD 240Гб, 500Вт, Mini-Tower – 1 шт. 7. Кондиционер HAIER HSU -24HPL03/R3 (Инв. № 210134000062198) – 1 шт. <p>Вешалка напольная (Инв. №1107-333144, Инв. №1107-333144) – 2 шт.</p>
<i>Центральная научная библиотека имени Н.И. Железнова</i>	Читальные залы библиотеки
<i>Студенческое общежитие</i>	Комната для самоподготовки

11. Методические рекомендации обучающимся по освоению дисциплины

В современных условиях творческая одаренность и нестандартная самостоятельная деятельность человека становятся основным ресурсом функционирования и развития общества. Процесс качественного обновления жизни в нашем обществе предполагает формирование устойчивого и долговременного спроса на творческую личность, яркую индивидуальность, на специалиста, свободно и критически мыслящего, самобытного и инициативного. Умение самостоятельно мыслить, свободно принимать решения, нести за них персональную ответственность необходимо молодежи еще и потому, что в современной жизни возросла автономия личности. И все же одним из важных требований социального заказа, предъявляемого выпускнику вуза в современных условиях, является умение самостоятельно пополнять свои знания, ориентироваться в стремительном потоке научной и культурной информации.

Промежуточным контролем по дисциплине является экзамен.

Организация самостоятельной работы обучающихся является одним из важнейших вопросов в условиях реализации компетентностной модели образования. Это связано не только с увеличением доли самостоятельной работы при освоении учебных дисциплин, но, прежде всего, с современным пониманием образования как жизненной стратегии личности. Мотивация к непрерывному образованию, общекультурные и профессиональные компетенции становятся необходимым ресурсом личности для успешного включения в трудовую деятельность и реализации своих жизненных планов. Основная задача высшего образования заключается в формировании творческой личности специалиста, способного к саморазвитию, самообразованию, инновационной деятельности.

Под самостоятельной работой обучающихся сегодня понимается вид учебно-познавательной деятельности по освоению основной образовательной программы высшего профессионального образования, осуществляемой в определенной системе, при партнерском участии преподавателя в ее планировании и оценке достижения конкретного результата.

Самостоятельная работа проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений обучающихся;
- углубления и расширения теоретических знаний;
- формирования умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развития познавательных способностей и активности обучающихся: творческой инициативы, самостоятельности, ответственности, организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, совершенствованию и самоорганизации;
- формирования общих и профессиональных компетенций;

- развитию исследовательских умений.

При выполнении заданий, вынесенных на самостоятельное изучение, необходимо наряду с библиотечным фондом пользоваться различными базами знаний, размещенными в Интернет, к которым, в частности, относятся: Научная электронная библиотека, Российская государственная библиотека и многие другие.

В подготовке к занятиям по дисциплине студенты должны активно использовать дополнительную литературу, поскольку именно с ее помощью можно получить наиболее полное и верное представление о происходящих в стране и в мире процессах. Для этих же целей необходимо шире использовать имеющиеся информационные технологии. Изучение литературы очень трудоемкая и ответственная часть подготовки к лабораторному занятию, написанию доклада и т.п. Она, как правило, сопровождается записями в той или иной форме. Конспектом называется краткая схематическая запись основного содержания научной работы. Желательно использование логических схем, делающих наглядным ход мысли конспектируемого автора.

Виды и формы отработки пропущенных занятий

Студент, пропустивший занятия обязан его отработать:

- лекцию отрабатывают путем устного ответа по пропущенной теме;
- практическое занятие путем выполнения практической работы, которая выполнялась на пропущенном практическом занятии, с разрешения преподавателя студент имеет право отработать пропущенное практическое задание самостоятельно и отчитаться по нему на ближайшем практическом занятии (если это не противоречит его плану) либо во время, назначенное преподавателем для индивидуальных консультаций.

12. Методические рекомендации преподавателям по организации обучения по дисциплине

В процессе обучения по дисциплине «Программная инженерия» используются лекционно-практические занятия, деловые игры, разбор конкретных ситуаций, проводятся дискуссии по актуальным проблемам управления, организуется работа с методическими и справочными материалами, целесообразно применение современных технических средств обучения и информационных технологий. Освоение учебной дисциплины предполагает осмысление её разделов и тем на практических занятиях, в процессе которых бакалавр должен закрепить и углубить теоретические знания.

Своеобразие современной профессиональной деятельности преподавателя заключается в необходимости ведения, поддержки и сопровождения студентов, что позволит сформировать новое поколение специалистов, обладающих **современными компетенциями**.

Дисциплина «Программная инженерия» имеет прикладной характер, её теоретические положения и практические навыки могут быть использованы в будущей практической деятельности.

Преподавание учебного материала по курсу целесообразно вести исходя из научно-обоснованных рекомендаций, с учетом преобразований, происходящих в экономике страны. Это система гибкого управления, способного своевременно перестраиваться и реагировать на конъюнктуру рынка, условия конкурентной борьбы и социальные факторы развития.

В процессе изучения дисциплины «Программная инженерия» предусмотрены несколько форм контроля: текущий и промежуточный.

Текущий контроль предназначен для определения качества усвоения лекционного материала. В течение учебного семестра рекомендуется назначать контрольные точки для проверки качества усвоения изучаемого материала по определенным темам в форме опроса, тестирования и выполнения заданий практикума по дисциплине.

Рекомендуется определять сроки проведения контрольных мероприятий, максимальная оценка за каждое из них и правила перевода общего количества баллов, полученных при изучении дисциплины, в промежуточный результат (экзамен).

Выполнение практических заданий является обязательным для всех обучающихся. Бакалавры, не выполнившие в полном объеме работы, предусмотренные учебным планом, не допускаются к сдаче экзамена.

Самостоятельная работа бакалавров по курсу должна обязательно сопровождаться проработкой конспекта, выполнением заданий и упражнений.

Программу разработал (и):

Демичев В.В., к.э.н., доцент,

Титов А.Д., ассистент



(подпись)



(подпись)

РЕЦЕНЗИЯ

на рабочую программу дисциплины Б1.О.31 «Программная инженерия» ОПОП ВО по направлению 09.03.02 Информационные системы и технологии, направ- ленности «Компьютерные науки и технологии искусственного интеллекта» (квалификация выпускника – бакалавр)

Чепуриной Екатериной Леонидовной, доцентом кафедры инженерной и компьютерной графики, кандидатом технических наук (далее по тексту рецензент), проведена рецензия рабочей программы дисциплины «Программная инженерия» ОПОП ВО по направлению 09.03.02 Информационные системы и технологии, направленности «Компьютерные науки и технологии искусственного интеллекта» (бакалавриат) разработанной в ФГБОУ ВО «Российский государственный аграрный университет – МСХА имени К.А. Тимирязева», на кафедре статистики и кибернетики (разработчики – Демичев Вадим Владимирович, доцент, к.э.н.; Титов Артем Денисович, ассистент).

Рассмотрев представленные на рецензирование материалы, рецензент пришел к следующим выводам:

1. Предъявленная рабочая программа дисциплины «Программная инженерия» (далее по тексту Программа) соответствует требованиям ФГОС ВО по направлению 09.03.02 Информационные системы и технологии. Программа содержит все основные разделы, соответствует требованиям к нормативно-методическим документам.

2. Представленная в Программе актуальность учебной дисциплины в рамках реализации ОПОП ВО не подлежит сомнению – дисциплина относится к формируемой участниками образовательных отношений части учебного цикла – Б1.О.

3. Представленные в Программе цели дисциплины соответствуют требованиям ФГОС ВО направления 09.03.02 Информационные системы и технологии.

4. В соответствии с Программой за дисциплиной «Программная инженерия» закреплено 3 компетенции (девять индикаторов). Дисциплина «Программная инженерия» и представленная Программа способна реализовать их в объявленных требованиях.

5. Общая трудоёмкость дисциплины «Программная инженерия» составляет 3 зачётных единицы (108 часов).

6. Информация о взаимосвязи изучаемых дисциплин и вопросам исключения дублирования в содержании дисциплин соответствует действительности. Дисциплина «Программная инженерия» взаимосвязана с другими дисциплинами ОПОП ВО и Учебного плана по направлению 09.03.02 Информационные системы и технологии и возможность дублирования в содержании отсутствует.

7. Представленная Программа предполагает использование современных образовательных технологий, используемые при реализации различных видов учебной работы. Формы образовательных технологий соответствуют специфике дисциплины.

8. Программа дисциплины «Программная инженерия» предполагает проведения занятий в интерактивной форме.

9. Виды, содержание и трудоёмкость самостоятельной работы студентов, представленные в Программе, соответствуют требованиям к подготовке выпускников, содержащимся во ФГОС ВО направления 09.03.02 Информационные системы и технологии. Представленные и описанные в Программе формы текущей оценки знаний (круглый стол как форма обсуждения отдельных вопросов, участие в дискуссиях, участие в тестировании, защита практических работ), соответствуют специфике дисциплины и требованиям к выпускникам.

Форма промежуточного контроля знаний студентов, предусмотренная Программой, осуществляется в форме экзамена, что соответствует статусу дисциплины, как дисциплины обязательной/вариативной части учебного цикла – Б1.О ФГОС ВО направления 09.03.02 Информационные системы и технологии.

10. Формы оценки знаний, представленные в Программе, соответствуют специфике дисциплины и требованиям к выпускникам.

11. Учебно-методическое обеспечение дисциплины представлено: основной литературой – 5 источник (базовый учебник), дополнительной литературой – 5 наименований, периодическими изданиями – 4 источников со ссылкой на электронные ресурсы, Интернет-ресурсы – 3 источника и соответствует требованиям ФГОС ВО направления *09.03.02 Информационные системы и технологии*.

12. Материально-техническое обеспечение дисциплины соответствует специфике дисциплины «Программная инженерия» и обеспечивает использование современных образовательных, в том числе интерактивных методов обучения.

13. Методические рекомендации студентам и методические рекомендации преподавателям по организации обучения по дисциплине дают представление о специфике обучения по дисциплине «Программная инженерия».

ОБЩИЕ ВЫВОДЫ

На основании проведенного рецензирования можно сделать заключение, что характер, структура и содержание рабочей программы дисциплины «Программная инженерия» ОПОП ВО по направлению *09.03.02 Информационные системы и технологии*, направленности «Компьютерные науки и технологии искусственного интеллекта» (квалификация выпускника – бакалавр), разработанная Демичевым Вадимом Владимировичем, доцент, к.э.н. и Титовым Артемом Денисовичем, ассистент соответствует требованиям ФГОС ВО, современным требованиям экономики, рынка труда и позволит при её реализации успешно обеспечить формирование заявленных компетенций.

Рецензент: Чепурина Е.Л., доцент инженерной и компьютерной графики ФГБОУ ВО «Российский государственный аграрный университет – МСХА имени К.А. Тимирязева», кандидат технических наук


(подпись)

«26» августа 2025 г.