

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Цели изучения дисциплины определены государственным образовательным стандартом высшего образования и соотнесены с общими целями ООП ВО по направлению подготовки 09.03.03 Прикладная информатика, в рамках которой преподается дисциплина. Цели дисциплины «Технологии создания и поддержки программного обеспечения»: формирование у студентов комплексных знаний и практических навыков в области современных технологий создания и поддержки программного обеспечения на протяжении всего жизненного цикла разработки (ЖЦР).

1.2 Задачи дисциплины

- Изучить основные модели ЖЦР ПО и их особенности.
- Освоить современные методологии разработки ПО, включая Agile-подходы.
- Научиться использовать системы контроля версий для совместной разработки.
- Приобрести навыки тестирования программного обеспечения на различных уровнях.
- Познакомиться с принципами автоматизации сборки, развертывания и мониторинга ПО (CI/CD).
- Освоить базовые концепции DevOps и их применение в практике разработки.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Технологии создания и поддержки программного обеспечения» относится к профильным дисциплинам, обеспечивающим формирование профессиональных компетенций выпускника по направлению подготовки 09.03.03 – Прикладная информатика.

Для успешного освоения дисциплины необходимы следующие пререквизиты: информатика, основы программирования, операционные системы и сети, технологии программирования.

Дисциплина «Технологии создания и поддержки программного обеспечения» является основой для изучения следующих дисциплин.

Постреквизиты: Анализ, проектирование и разработка БД, Облачные технологии и бэкэнд-разработка, Разработка мобильных приложений, а также других дисциплин связанных с: разработкой, тестированием, поддержкой программного обеспечения, разработкой архитектур информационных систем. Углубляет знания, полученные в рамках дисциплин «Основы программирования, Объектно-ориентированное программирование», «Разработка пользовательского WEB интерфейса, Основы программирования на Python».

1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Роль 1: Data Analyst (Аналитик данных)

Задачи:

1. Статистический анализ, визуализация данных, предварительная обработка.
2. Создание прогнозных моделей
3. Построение аналитических моделей для поддержки бизнес-решений.

Роль 2: MLOps (Специалист по эксплуатации ИИ)

Задачи:

1. DevOps для ML.
2. Автоматизация, мониторинг ML-систем.

3. Операционное управление жизненным циклом ML-моделей.

Роль 3: AI PM (Менеджер проектов ИИ)

Задачи:

1. Управление ИИ-проектами от идеи до внедрения
2. Анализ бизнес-требований и постановка задач
3. Оценка эффективности и ROI ИИ-решений

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

Код и наименование индикатора*	Результаты обучения по дисциплине
ПК-3 Способен участвовать в разработке, сопровождении и администрировании информационных систем и программного обеспечения	
ПК-3.1 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	Знает основные модели и методологии жизненного цикла ПО (каскадная, V-модель, Agile). Умеет выбирать и обосновывать выбор модели ЖЦП для конкретного проекта.
ПК-3.2 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла	Владеет системой контроля версий Git на уровне организации репозитория, ветвления (GitFlow), слияния и разрешения конфликтов. Применяет практики командной работы и code review в рамках общего репозитория.
ПК-3.3 Способен управлять процессами жизненного цикла ИТ-продукта	Умеет настраивать процессы непрерывной интеграции и доставки (CI/CD) с использованием современных инструментов (напр., GitHub Actions). Применяет принципы DevOps для автоматизации сборки, тестирования и развертывания ПО.
ЛС-4 Способен управлять процессом жизненного цикла ИИ-продукта	
ЛС-4.1 Осуществляет запуск и ведение проекта в области ИИ, в том числе планирование и контроль задач, оценку ресурсов	Подбирает методологию управления проектами (Agile, Scrum, Kanban) под ограничения задачи и ресурсное обеспечение. Осуществляет декомпозицию проекта на задачи, оценивает трудозатраты и строит рабочий план (бэклог) в инструментах управления (напр., Jira).
ЛС-4.2 Координирует и контролирует работу команд проекта с целью достижения общих целей проекта	Демонстрирует эффективное владение инструментами коммуникаций и проектного управления. Координирует взаимодействие между техническими и бизнес-командами в рамках скрам-митингов и итерационных обзоров. Применяет базовые принципы управления стейкхолдерами и работы с ожиданиями.

Результаты обучения по дисциплине достигаются в рамках осуществления всех видов контактной и самостоятельной работы обучающихся в соответствии с утвержденным учебным планом.

Индикаторы достижения компетенций считаются сформированными при достижении соответствующих им результатов обучения.

Содержание дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы (темы) дисциплины, изучаемые в 4 семестре (очная форма обучения)

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная Работа			Внеаудиторная работа
			Л	ПЗ	ЛР	

1.	Жизненный цикл разработки программного обеспечения (ЖЦР)	17	4		8	5
2.	Методологии разработки ПО	17	4		8	5
3.	Системы контроля версий (Git)	17	4		8	5
4.	Тестирование программного обеспечения	18,8	4		8	6,8
	ИТОГО по разделам дисциплины	69,8	16		32	21,8
	Контроль самостоятельной работы (КСР)	2				
	Промежуточная аттестация (ИКР)	0,2				
	Подготовка к текущему контролю					
	Общая трудоемкость по дисциплине	72				

Примечание: Л – лекции, ПЗ – практические занятия / семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

Разделы (темы) дисциплины, изучаемые в 5 семестре (очная форма обучения)

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная Работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1.	Автоматизация сборки и развертывания ПО (CI/CD)	17	4		8	5
2.	Основы DevOps	18	4		9	5
3.	Контейнеризация (Docker)	18	4		9	5
4.	Оркестрация контейнеров (Kubernetes)	16,8	4		8	4,8
	ИТОГО по разделам дисциплины	69,8	16		34	19,8
	Контроль самостоятельной работы (КСР)	2				
	Промежуточная аттестация (ИКР)	0,2				
	Подготовка к текущему контролю					
	Общая трудоемкость по дисциплине	72				

Примечание: Л – лекции, ПЗ – практические занятия / семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента