Приложение № 4
к образовательной программе высшего образования – программе магистратуры
по направлению подготовки 09.04.01 Информатика и вычислительная техника
(направленность (профиль) «Разработка и управление
критической информационной инфраструктурой»)

**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ «НАУЧНО-ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ «СИРИУС»
(АНОО ВО «УНИВЕРСИТЕТ «СИРИУС»)**

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 7835bcdac1c89a3da835a18f6045494f3f404337
Владелец: Некрасова Людмила Андреевна
Действителен: с 08.07.2026 до 08.07.2027

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«Основы безопасности при разработке и внедрении программного обеспечения»

1. Трудоемкость дисциплины (модуля): 3 з.е.

2. Место дисциплины (модуля) в учебном плане:

Обязательная часть «Профессиональной подготовки» в рамках Блока 1. Дисциплины (модули), реализуется во 2 семестре.

3. Цель дисциплины (модуля): изучение методов обеспечения безопасности в процессе разработки и внедрения программного обеспечения; знакомство с основами проектирования, реализации безопасных приложений, процессом аутентификации и авторизации, а также углубление знаний в области криптографии и методов обеспечения безопасности приложений.

4. Задачи дисциплины (модуля):

– сформировать представление о принципах DevSecOps, CI/CD и безопасной контейнеризации;

– обеспечить владение навыками настройки Git, пайплайнов CI/CD и инструментов анализа кода;

– научить применять инфраструктуру как код и методологию Twelve-Factor App при разработке защищённого ПО.

5. Перечень разделов (тем) дисциплины (модуля) и их краткое содержание:

Наименование раздела (темы) дисциплины (модуля)	**Краткое содержание**
Раздел 1. Системы контроля версий и автоматизация сборки приложений	1.1. Жизненный цикл ПО. 1.2. DevOps и Git. 1.3. Фреймворки для автоматизации сборки проекта. 1.4. Виртуализация и контейнеризация: Docker, Docker-compose, альтернативы Docker. 1.5. Хранение и передача чувствительных данных.
Раздел 2. Continuous Integration, Continuous Delivery и Continuous Deployment	2.1. Continuous Integration: хранение кода, сборка, тестирование. 2.2. GitLab CI. GitHub Actions. 2.3. SAST. DAST. Сканеры безопасности кода. 2.4. Системы хранения артефактов. 2.5. Организация процесса поставки.
Раздел 3. Инфраструктура как код	3.1. Методология Twelve-Factor App. 3.2. Организация резервного копирования. 3.3. Infrastructure as Code.

6. Образовательные результаты освоения дисциплины (модуля):

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
ПК-2. Способен разрабатывать и внедрять программные продукты для объектов критической информационной инфраструктуры	ИПК-2.1. Знает основы синтаксиса высокоуровневых и низкоуровневых языков программирования, а также знаком с паттернами разработки программного обеспечения как для микроэлектроники, так и для микросервисных приложений
	ИПК-2.2. Умеет разрабатывать программные продукты с использованием различных языков программирования и внедрять их в объекты критической инфраструктуры на аппаратном и программном уровнях
	ИПК-2.3. Владеет методами создания, внедрения, тестирования и обеспечения безопасности программного обеспечения на всех этапах жизненного цикла продукта, в том числе с применением технологий искусственного интеллекта, а также способен разработать программное обеспечение как для микроэлектроники, так и для микросервисных и монолитных систем
ПК-5. Способен анализировать и реагировать на угрозы информационной безопасности	ИПК-5.1. Знает основы анализа и классификации угроз информационной безопасности, виды вредоносного программного обеспечения
	ИПК-5.2. Умеет выявлять и устранять угрозы информационной безопасности, а также обнаруживать и реагировать на вредоносное программное обеспечение
	ИПК-5.3. Владеет методами оперативного реагирования на угрозы, включая обнаружение и устранение вредоносного программного обеспечения, а также способен создавать собственные методологии для обеспечения защиты информационной безопасности