

Министерство науки и высшего образования РФ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.В.02 Программная инженерия

наименование дисциплины (модуля) в соответствии с учебным планом

Направление подготовки / специальность

09.04.01 Информатика и вычислительная техника

Направленность (профиль)

09.04.01.13 Инженерия искусственного интеллекта

Форма обучения

очная

Год набора

2022

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программу составили _____

Ст. преподаватель, Пахомова Кристина Игоревна

должность, инициалы, фамилия

1 Цели и задачи изучения дисциплины

1.1 Цель преподавания дисциплины

Целью освоения дисциплины «Программная инженерия» является представление программной инженерии в виде целостного изложения, освещающая концепцию процесса, различные методологии разработки программного обеспечения, отличие программной инженерии от других отраслей. Студент в ходе обучения учится оперировать профессиональными терминами и формирует представление о специфике профессии.

1.2 Задачи изучения дисциплины

Задачи рассматривания дисциплины - изучение основных подходов к организации командной разработки систем машинного обучения и искусственного интеллекта, современных технологий разработки программного обеспечения, процессов командной разработки ПО, анализу формальных и гибких технологий разработки ПО, способов обеспечения качества программных продуктов и мотивации членов команды разработки.

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

Код и наименование индикатора достижения компетенции	Запланированные результаты обучения по дисциплине
ПК-1: Способен исследовать и разрабатывать архитектуры систем искусственного интеллекта для различных предметных областей на основе комплексов методов и инструментальных средств систем искусственного интеллекта	
ПК-1.1: Исследует и разрабатывает архитектуры систем искусственного интеллекта для различных предметных областей	Современные методы построения архитектуры в области разработки систем искусственного документа в разных сферах Реализовать архитектуру информационных систем в области искусственного интеллекта Навыками организации взаимодействия компонентов архитектуры информационных систем в области искусственного интеллекта
ПК-1.2: Выбирает комплексы методов и инструментальных средств искусственного интеллекта для решения задач в зависимости от особенностей предметной области	Набор инструментов и методов для разработки моделей искусственного интеллекта Использовать необходимые инструменты для разработки моделей искусственного интеллекта из множества существующих Навыками интегрирования инструментов для разработки моделей искусственного интеллекта в существующие и новые проекты

<p>ПК-1.3: Разрабатывает единые стандарты в области безопасности (в том числе отказоустойчивости) и совместимости программного</p>	<p>Нормативные документы информационной безопасности в области разработки программного обеспечения с использованием искусственного интеллекта Применять эталонные архитектуры вычислительных</p>
<p>обеспечения, эталонных архитектур вычислительных систем и программного обеспечения, а также определение критериев сопоставления программного обеспечения и критериев эталонных открытых тестовых сред (условий) в целях определения качества и эффективности программного обеспечения технологий и систем искусственного интеллекта</p>	<p>систем в разработке программного обеспечения Методами расчетов критериев качества для информационных систем</p>
<p>ПК-2: Способен выбирать, разрабатывать и проводить экспериментальную проверку работоспособности программных компонентов систем, основанных на знаниях, по обеспечению требуемых критериев эффективности и качества функционирования</p>	
<p>ПК-2.1: Выбирает и разрабатывает программные компоненты систем, основанных на знаниях</p>	<p>Знает основные критерии эффективности и качества функционирования системы, основанной на знаниях: точность, релевантность, достоверность, целостность, быстрота решения задач, надежность, защищенность функционирования систем, основанных на знаниях Выбирать и разрабатывать программные компоненты систем, основанные на знаниях Навыками выбира и разработки программных компонент систем, основанные на знаниях</p>
<p>ПК-2.2: Проводит экспериментальную проверку работоспособности систем, основанных на знаниях</p>	<p>Алгоритмы и технологии проверки работоспособности систем, основанных на знаниях Проводить экспериментальную проверку работоспособности систем, основанных на знаниях Навыками проведения экспериментальной проверки работоспособности систем, основанных на знаниях</p>
<p>ПК-8: Способен разрабатывать и модернизировать программное и аппаратное обеспечение технологий и систем искусственного интеллекта с учетом требований информационной безопасности в различных предметных областях</p>	

ПК-8.1: Разрабатывает программное и аппаратное обеспечение технологий и систем искусственного интеллекта для решения профессиональных задач с	Нормативные документы, связанные с разработкой программного обеспечения в сфере информационной безопасности Программировать на языках высокого уровня Навыками написания программного кода с использованием сред разработки
учетом требований информационной безопасности в различных предметных областях	
ПК-8.2: Модернизирует программное и аппаратное обеспечение технологий и систем искусственного интеллекта для решения профессиональных задач с учетом требований информационной безопасности в различных предметных областях	Актуальные нормативные документы, связанные с разработкой программного обеспечения в сфере информационной безопасности Программировать на языках высокого уровня в соответствии с техническим заданием проекта Навыками написания программного кода с использованием современных и актуальных сред разработки

1.4 Особенности реализации дисциплины

Язык реализации дисциплины: Русский.

Дисциплина (модуль) реализуется с применением ЭО и ДОТ

URL-адрес и название электронного обучающего курса: <https://e.sfu-kras.ru/course/view.php?id=34333>.

2. Объем дисциплины (модуля)

Вид учебной работы	Всего, зачетных единиц (акад. час)	е
		1
Контактная работа с преподавателем:	1 (36)	
занятия лекционного типа	0,5 (18)	
практические занятия	0,5 (18)	
Самостоятельная работа обучающихся:	2 (72)	
курсовое проектирование (КП)	Нет	
курсовая работа (КР)	Нет	

3 Содержание дисциплины (модуля)

3.1 Разделы дисциплины и виды занятий (тематический план занятий)

		Контактная работа, ак. час.							
№ п/п	Модули, темы (разделы) дисциплины	Занятия лекционного типа		Занятия семинарского типа				Самостоятельная работа, ак. час.	
				Семинары и/или Практические занятия		Лабораторные работы и/или Практикумы			
		Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС
1. Введение в программную инженерию.									
	1. Отличие программы от программного продукта. Software Engineering Body of Knowledge. Управление требованиями к программному обеспечению. DevOps. MLOps. Жизненный цикл приложений машинного обучения.	2							
	2. Инструменты командной разработки. Система контроля версий Git. Сервис GitHub. Основы работы с Git в командной строке.			2					
	3. Цели тестирования программного обеспечения. Виды тестирования. Модульное тестирование.	2							
	4. Модульное тестирование в Python: pytest.			2					
	5. Основы Continuous Integration (CI).	2							
	6. Инструменты Continuous Integration. Continuous Integration на GitHub.			2					

7. Продвинутый уровень командной разработки.							10	
8. Ветки (branches) в репозиториях программного кода. Предложения по изменению кода (pull request).			2					
9. Понятие качества кода. Зачем нужен чистый код. Рефакторинг.	2							
10. Инструменты для рефакторинга.			2					
11. Рецензирование кода (Code Review)	2							
12. Назначение Code Review. Лучшие практики Code Review. Code Review на GitHub. Человеческий фактор в Code Review.			2					
13. Жизненный цикл программного продукта.	2							
14. Руководство по стилю в Python PEP 8. Форматтеры кода (в IDE, YAPF, Black). Линтеры (Flacke8, Pylint).							26	
2. Архитектура программного обеспечения								
1. Организация работы приложения машинного обучения через API.	4							
2. Инструменты для разработки API: FastAPI, Flask. Организации доступа к модели машинного обучения через API.			4					
3. Создание пайплайнов приложений машинного обучения.	2							
4. Обучение модели. Развертывание модели. Необходимость автоматизации пайплайнов.			2					
5. Проектирование кода для повторного использования. Модули и пакеты в Python. Библиотеки в Python. Создание собственных библиотек в Python.							36	

Bcero	18		18				72	
-------	----	--	----	--	--	--	----	--

4 Учебно-методическое обеспечение дисциплины

4.2 Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства (программное обеспечение, на которое университет имеет лицензию, а также свободно распространяемое программное обеспечение):

1. Используется бесплатно-распространяемое программное обеспечение:
2. Язык Python – <https://www.python.org/>
3. Система контроля версий Git – <https://git-scm.com>
4. GitHub – <https://github.com/>
5. Форматтер YAPF – <https://github.com/google/yapf>
6. Форматтер Black – <https://github.com/psf/black>
7. Линтер Flake8 – <https://github.com/pycqa/flake8>
8. Линтер Pylint – <https://github.com/PyCQA/pylint/>
9. Библиотека машинного обучения Hugging Face <https://huggingface.co>
10. Облачная платформа <https://www.heroku.com/>
11. FastAPI – <https://fastapi.tiangolo.com/>
12. Data Version Control – <https://dvc.org/>

4.3 Интернет-ресурсы, включая профессиональные базы данных и информационные справочные системы:

1. Электронные ресурсы (издания):
2. Software Engineering Body of Knowledge (SWEBOOK). URL: <https://www.computer.org/education/bodies-of-knowledge/software-engineering> (дата обращения: 05.10.2021).
3. GitHub Actions. URL: <https://docs.github.com/en/actions> (дата обращения: 05.10.2021).
4. Software Engineering at Google. <https://abseil.io/resources/swe-book> (дата обращения: 05.10.2021).
5. Scott Chacon, Ben Straub. Pro Git. <https://git-scm.com/book/ru/v2> (дата обращения: 05.10.2021).
6. Журнал "Программная инженерия". URL: <http://novtex.ru/prin/rus/> (дата обращения: 05.10.2021).
- 7.
- 8.
9. Профессиональные базы данных, информационно-справочные системы
- 10.
11. Цифровая библиотека научно-технических изданий Института инженеров по электротехнике и радиоэлектронике (Institute of Electrical and Electronic Engineers (IEEE)) на английском языке – <http://www.ieee.org/ieeexplore>

12. Oxford University Press – <http://www.oxfordjournals.org/en/>
13. Архив препринтов с открытым доступом – <https://arxiv.org/>

5 Фонд оценочных средств

Оценочные средства находятся в приложении к рабочим программам дисциплин.

6 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине (модулю)

Специальные помещения должны представлять собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Специальные помещения должны быть укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории. Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, обеспечивающие тематические иллюстрации, рабочим учебным программам дисциплин (модулей). Помещения для самостоятельной работы обучающихся должны быть оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации.