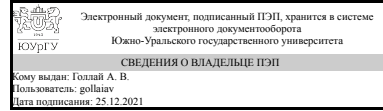


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Директор института
Высшая школа электроники и
компьютерных наук



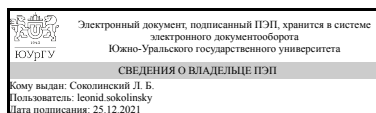
А. В. Голлай

РАБОЧАЯ ПРОГРАММА

дисциплины 1.О.05 Современные методы DevOps
для направления 09.04.04 Программная инженерия
уровень Магистратура
форма обучения очная
кафедра-разработчик Системное программирование

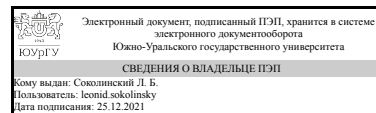
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.04.04 Программная инженерия, утверждённым приказом Минобрнауки от 19.09.2017 № 932

Зав.кафедрой разработчика,
д.физ.-мат.н., проф.



Л. Б. Соколинский

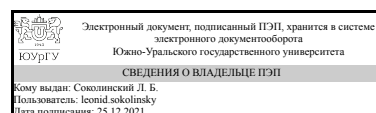
Разработчик программы,
д.физ.-мат.н., проф., заведующий
кафедрой



Л. Б. Соколинский

СОГЛАСОВАНО

Руководитель направления
д.физ.-мат.н., проф.



Л. Б. Соколинский

1. Цели и задачи дисциплины

Целями освоения дисциплины является изучение современных принципов, методов и инструментов DevOps в процессах жизненного цикла разработки ПО. Основными задачами дисциплины являются: освоение на практике инструментов DevOps для решения задач сборки, непрерывной интеграции, мониторинга, оркестрации, журналирования, обеспечения информационной безопасности в проектах разработки ПО.

Краткое содержание дисциплины

В рамках освоения дисциплины будут изучены основные принципы, методы, паттерны и антипаттерны DevOps. Научатся применять на практике инструменты контейнеризации, оркестрации, непрерывной интеграции, мониторинга в процессах разработки ПО. Отдельно будут рассмотрены инструменты MLOps для обеспечения версионирования данных, воспроизводимости экспериментов в проектах с применением моделей и алгоритмов машинного обучения.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
УК-3 Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели	Знает: методы командной разработки ПО с применением методологии DevOps Умеет: применять инструменты DevOps в работе команды разработки с целью реализации практик непрерывной интеграции и поставки ПО Имеет практический опыт: применения технологий MLOps в проектах разработки программных систем, в том систем искусственного интеллекта
ОПК-8 Способен осуществлять эффективное управление разработкой программных средств и проектов	Знает: основные принципы методологии DevOps при управлении разработкой ПО Умеет: управлять процессами интеграции, развертывания и поставки ПО в проектах с использованием технологий DevOps Имеет практический опыт: использования инструментов DevOps
ПК-1 Способен проводить анализ требований к архитектуре программного обеспечения, осуществлять выбор и моделирование архитектуры единой информационной системы, осуществлять документирование программного обеспечения, контролировать реализацию и тестирование программного обеспечения	Знает: принципы и инструменты MLOps - применения технологий DevOps при разработке систем искусственного интеллекта Умеет: автоматизировать процессы интеграции и развертывания моделей машинного обучения с использованием инструментов MLOps Имеет практический опыт: применения технологий MLOps в проектах разработки систем искусственного интеллекта
ПК-12 Способен исследовать и разрабатывать архитектуры систем искусственного интеллекта для различных предметных областей на основе комплексов методов и инструментальных средств систем искусственного интеллекта	Знает: методы и инструментальные средства систем искусственного интеллекта, критерии их выбора и методы комплексирования в рамках создания интегрированных гибридных интеллектуальных систем различного назначения

	Умеет: выбирать и применять технологии DevOps на основе анализа требований, контролировать процессы интеграции и поставки для повышения качества ПО, сокращения времени выпуска стабильных релизов ПО
--	---

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
1.Ф.04 Объектно-ориентированные CASE-технологии, 1.О.04 Архитектура распределенных вычислительных систем, 1.Ф.05 Разработка интеллектуальных систем на языке R	Не предусмотрены

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
1.Ф.05 Разработка интеллектуальных систем на языке R	Знает: методы и критерии оценки качества моделей машинного обучения Умеет: выбирать, применять и интегрировать методы и инструментальные средства систем искусственного интеллекта, критерии их выбора и методы комплексирования в рамках создания интегрированных гибридных интеллектуальных систем различного назначения Имеет практический опыт:
1.Ф.04 Объектно-ориентированные CASE-технологии	Знает: основные особенности процесса проектирования программных систем, типы черт программных систем (поведенческие, структурные), классификацию моделей UML, основные виды диаграмм UML, понятия, использующиеся в метаязыке UML и в конкретных видах диаграмм, архитектурные принципы построения систем искусственного интеллекта, методы декомпозиции основных подсистем (компонентов) и реализации их взаимодействия на основе методологии предметно-ориентированного проектирования Умеет: выделять функциональные требования к разрабатываемой системе, определять поведенческие и структурные черты проектируемого ПО, строить модели проектируемого продукта с помощью различного типа диаграмм UML Имеет практический опыт: навыками проектирования структуры и поведения программных систем, навыками анализа предметной области, спецификации поведенческих и структурных черт разрабатываемой информационной системы,

	оформления документации на этапе проектирования системы
1.О.04 Архитектура распределенных вычислительных систем	<p>Знает: Современные концепции проектирования распределенных вычислительных систем на основе клиентсерверного, однорангового и сервисориентированного подходов а также очередей сообщений Умеет: выстраивать архитектуру системы искусственного интеллекта, осуществлять декомпозицию основных подсистем (компонентов) и реализации их взаимодействия на основе методологии предметно-ориентированного проектирования, Разрабатывать приложения на основе клиентсерверного и сервис-ориентированного подходов, а также приложения с использованием очередей сообщений, осуществлять осознанный выбор технологии сериализации данных для обеспечения коммуникации между компонентами распределенного приложения Имеет практический опыт: Создания приложений на основе технологии gRPC и концепции REST</p>

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 2 з.е., 72 ч., 38,25 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах	
		Номер семестра	
		3	
Общая трудоёмкость дисциплины	72	72	
<i>Аудиторные занятия:</i>	32	32	
Лекции (Л)	16	16	
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	16	16	
Лабораторные работы (ЛР)	0	0	
<i>Самостоятельная работа (СРС)</i>	33,75	33,75	
с применением дистанционных образовательных технологий	0		
Подготовка к зачету	8	8	
Изучение основной и дополнительной литературы	25,75	25.75	
Консультации и промежуточная аттестация	6,25	6,25	
Вид контроля (зачет, диф.зачет, экзамен)	-		зачет

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР

1	Основные принципы и инструменты DevOps	20	10	10	0
2	Основные принципы и инструменты MLOps (Machine Learning Operations)	8	4	4	0
3	Методы DevOps в жизненном цикле разработки ПО	4	2	2	0

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов
1	1	Введение: Что такое DevOps и в чем его ценность. Профессия DevOps-инженер: роли и функции. Система сборки - контейнеризация (Docker). Управление артефактами (Docker Registry). Оркестрация (Kubernetes)	4
2	1	Continuous Integration (Gitlab CI). Управление конфигурациями (Ansible).	2
3	1	Информационная безопасность (GPG, Secrets, Vault, SSL)	2
4	1	Мониторинг (Grafana, Zabbix, Prometheus). Журналирование (Logstash)	2
5	2	Специфика конфигураций и процессов разработки в проектах с ML-моделями. Основы MLOps. Версионирование данных (DVC). Воспроизводимость экспериментов на данных (MLflow).	2
6	2	Выстраивание и автоматизация pipeline в ML-проектах. Настройка CI/CD в ML-проектах. Мониторинг работы ML-моделей с Grafana	2
7	3	Методы DevOps в жизненном цикле разработки ПО. Инцидент-менеджмент. Документация как один из элементов коммуникации (Confluence). Топологии DevOps (паттерны и антипаттерны)	2

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	1	Практика с контейнеризацией в Docker. Управление артефактами с Docker Registry.	2
2	1	Настройка оркестрации с помощью Kubernetes	2
3	1	Настройка конвейера CI/CD в Gitlab CI.	2
4	1	Применение Ansible для управления конфигурациями. Настройка безопасности в веб-проекте (GPG, Secrets, Vault, SSL).	2
5	1	Мониторинг сервисов с использованием Prometheus. Журналирование с помощью Logstash	2
6	2	Применение версионирования данных с DVC. Обеспечение воспроизводимости экспериментов на данных с MLflow.	2
7	2	Выстраивание и автоматизация pipeline в ML-проектах. Оркестрация с Kubernetes в ML-проектах. Настройка CI/CD в ML-проектах. Мониторинг работы ML-моделей с Grafana	2
8	3	Практика по завершению внедрения DevOps в живой проект разработки ПО	2

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

Выполнение СРС

Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Подготовка к зачету	Основная литература 1, 2. Дополнительная литература 1-4	3	8
Изучение основной и дополнительной литературы	Основная литература 1, 2. Дополнительная литература 1-4	3	25,75

6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Се-местр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учи-тыва-ется в ПА
1	3	Текущий контроль	ПЗ-1. Контейнеризация в Docker. Управление артефактами с Docker Registry.	1	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%, 0 баллов: задание не выполнено	зачет
2	3	Текущий контроль	ПЗ-2. Настройка оркестрации с помощью Kubernetes	2	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%, 0 баллов: задание не выполнено	зачет
3	3	Текущий контроль	ПЗ-3. Настройка конвейера CI/CD в Gitlab CI.	1	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%,	зачет

						0 баллов: задание не выполнено	
4	3	Текущий контроль	ПЗ-4. Применение Ansible для управления конфигурациями. Настройка безопасности в веб-проекте (GPG, Secrets, Vault, SSL).	2	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%, 0 баллов: задание не выполнено	зачет
5	3	Текущий контроль	ПЗ-5. Мониторинг сервисов с использованием Prometheus. Журналирование с помощью Logstash	2	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%, 0 баллов: задание не выполнено	зачет
6	3	Текущий контроль	ПЗ-6. Применение версионирования данных с DVC. Обеспечение воспроизводимости экспериментов на данных с MLflow.	2	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%, 0 баллов: задание не выполнено	зачет
7	3	Текущий контроль	ПЗ-7. Выстраивание и автоматизация pipeline в ML-проектах.	3	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%, 0 баллов: задание не выполнено	зачет
8	3	Текущий контроль	ПЗ-8. Завершение внедрения DevOps в живой проект разработки ПО	3	3	3 балла: задание выполнено полностью, 2 балла: задание выполнено полностью, но допущены незначительные ошибки, или задание выполнено более, чем 50%, 1 балла: задание выполнено полностью, но допущены серьезные ошибки, или задание выполнено менее, чем 50%,	зачет

						0 баллов: задание не выполнено	
9	3	Промежуточная аттестация	Итоговый тест	1	15	Компьютерный тест состоит из 15 вопросов, позволяющих оценить сформированность компетенций. На ответы отводится 1 час. 15 баллов: задание полностью выполнено без ошибок 1-14 баллов: задание выполнено частично или выполнено с ошибками 0 баллов: задание не выполнено	зачет

6.2. Процедура проведения, критерии оценивания

Вид промежуточной аттестации	Процедура проведения	Критерии оценивания
зачет	На зачете происходит оценивание учебной деятельности обучающихся по дисциплине на основе полученных оценок за контрольно-рейтинговые мероприятия текущего контроля и промежуточной аттестации. При оценивании результатов учебной деятельности обучающегося по дисциплине используется балльно-рейтинговая система оценивания результатов учебной деятельности обучающихся (утверждена приказом ректора от 24.05.2019 г. № 179) Зачтено: Величина рейтинга обучающегося по дисциплине 60...100 % Незачтено: Величина рейтинга обучающегося по дисциплине 0...59 %. Допускается выставление оценки на основе текущего рейтинга (автоматом).	В соответствии с пп. 2.5, 2.6 Положения

6.3. Паспорт фонда оценочных средств

Компетенции	Результаты обучения	№ КМ								
		1	2	3	4	5	6	7	8	9
УК-3	Знает: методы командной разработки ПО с применением методологии DevOps	+				+				+
УК-3	Умеет: применять инструменты DevOps в работе команды разработки с целью реализации практик непрерывной интеграции и поставки ПО	+	+	+	+					+
УК-3	Имеет практический опыт: применения технологий MLOps в проектах разработки программных систем, в том систем искусственного интеллекта	+					+	+		+
ОПК-8	Знает: основные принципы методологии DevOps при управлении разработкой ПО	+								+
ОПК-8	Умеет: управлять процессами интеграции, развертывания и поставки ПО в проектах с использованием технологий DevOps		+	+	+					+
ОПК-8	Имеет практический опыт: использования инструментов DevOps	+	+	+						+
ПК-1	Знает: принципы и инструменты MLOps - применения технологий DevOps при разработке систем искусственного интеллекта						+	+		+
ПК-1	Умеет: автоматизировать процессы интеграции и развертывания моделей машинного обучения с использованием инструментов MLOps						+	+		+
ПК-1	Имеет практический опыт: применения технологий MLOps в проектах разработки систем искусственного интеллекта						+	+		+

			https://e.lanbook.com/book/116130 . — Режим доступа: для авториз. пользователей..
4	Дополнительная литература	Электронно-библиотечная система издательства Лань	Лукша, М. Kubernetes в действии / М. Лукша ; перевод с английского А. В. Логунов. — Москва : ДМК Пресс, 2019. — 672 с. — ISBN 978-5-97060-657-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/131688 . — Режим доступа: для авториз. пользователей.
5	Дополнительная литература	Электронно-библиотечная система издательства Лань	Кочер, П. С. Микросервисы и контейнеры Docker : руководство / П. С. Кочер ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2019. — 240 с. — ISBN 978-5-97060-739-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/123710 . — Режим доступа: для авториз. пользователей..
6	Дополнительная литература	Электронно-библиотечная система издательства Лань	Хохштейн, Л. Запускаем Ansible. Простой способ автоматизации управления конфигурациями и развертыванием приложения / Л. Хохштейн, Р. Мозер ; перевод с английского Е. В. Филонова, А. Н. Киселев. — Москва : ДМК Пресс, 2018. — 382 с. — ISBN 978-5-97060-513-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/108128 . — Режим доступа: для авториз. пользователей.

Перечень используемого программного обеспечения:

Нет

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

Вид занятий	№ ауд.	Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий
Лекции	434 (36)	Компьютер и проектор.
Практические занятия и семинары	114-1 (2)	Компьютерный класс, имеется выход в интернет
Зачет, диф. зачет	114-1 (2)	Компьютерный класс, имеется выход в интернет