ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНА

Приказом ректора
АНО ВО «Центральный университет»
от «02» июня 2026 г. № 0602.30

Рабочая программа дисциплины (модуля)
«Основы промышленной разработки»
дополнительной профессиональной программы – программы профессиональной переподготовки «Машинное обучение в индустрии»

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 0215B0C000D1B3FEB44A37EE631E0C27D4
Владелец: Ивашкевич Евгений Васильевич
Действителен: с 13.01.2026 до 13.04.2027

Электронный документ

Москва
2026

Содержание

1. **Краткая характеристика дисциплины (модуля)** 3
2. **Тематический план** 4
3. **Содержание дисциплины (модуля)** 5
4. **Учебно-методическое обеспечение** 6
5. **Материально-техническое обеспечение** 6
6. **Методические и оценочные материалы** 8

1. Краткая характеристика дисциплины (модуля)

Изучение дисциплины (модуля) «Основы промышленной разработки» обеспечивает слушателей пониманием ключевых процессов и методов, используемых в промышленной разработке, что является основой для успешной карьеры в инженерных и технических областях. Освоение основ промышленной разработки способствует развитию критического мышления и способности к решению сложных задач.

Цель изучения дисциплины (модуля): формирование у слушателей знаний и навыков для эффективной разработки, тестирования и интеграции веб-приложений в командной среде с использованием клиент-серверной архитектуры и баз данных.

Задачи изучения дисциплины (модуля):

- изучить принципы организации и управления процессами командной разработки программного обеспечения;
- освоить методы декомпозиции сложных проектов на технологические компоненты и задачи;
- научиться создавать и применять различные виды тестирования для обеспечения качества кода;
- развить умения программного взаимодействия с базами данных для хранения и обработки информации;
- приобрести навыки интеграции индивидуальных модулей в общий проект с учетом особенностей клиент-серверных приложений.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- механизмы клиент-серверного взаимодействия приложений;
- процессы разработки в команде.

уметь:

- интегрировать свою часть кода в командный проект;
- покрывать код различными видами тестов;
- взаимодействовать с базами данных программным образом.

владеть:

- навыком декомпозиции проекта по используемым технологиям и подзадачам;
- навыком реализации полноценного веб-приложения с хранением данных в базе данных.

2. Тематический план

№ п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостоятельная работа	
		Лекции	Семинары (практические занятия)			
1	Основы ОС Linux, работа с командной строкой	2	2		11	Домашние задания
2	Система контроля версий Git	2	2		11	Домашние задания
3	Продвинутая работа с Bash	2	2		11	Домашние задания
4	Система контроля версий Git	3	2		11	Домашние задания
5	Работа с удаленным сервером по SSH	3	2		11	Домашние задания
6	Организация проекта	3	3		11	Домашние задания
7	Основы сетевого взаимодействия	3	3	4	11	Контрольная работа
8	Сетевое взаимодействие	3	3		11	Домашние задания
9	Контейнеризация и Docker	3	3		12	Домашние задания
10	Непрерывная интеграция/Непрерывная доставка	3	3		12	Домашние задания
11	Принципы написания программного кода	3	3		12	Домашние задания
	Зачет с оценкой			4		
	Итого:	***30***	***28***	***8***	***124***	
Объем дисциплины (модуля) (в ак. ч.)		***190***				
Объем дисциплины (модуля) (в зач. ед.)		***5***				

3. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Основы ОС Linux, работа с командной строкой	Установка Linux. Командная строка и основные команды. Устройство файловой системы. Права доступа. Процессы.
2	Система контроля версий Git	Введение. Репозитории, коммиты, ветки, состояния файлов. Удалённые репозитории и синхронизация с ними.
3	Продвинутая работа с Bash	Переменные, операторы. Конвейеры. Продвинутые команды.
4	Система контроля версий Git	Отмена изменений. Стратегии слияния. Git Workflows.
5	Работа с удаленным сервером по SSH	Удалённое подключение к серверам. SSH, SCP, FTP, SFTP. Работа через VNC.
6	Организация проекта	Виды систем сборок. Механизмы работы систем сборки. Сборка Python-проектов: pybind11, пакетные менеджеры poetry, uv.
7	Основы сетевого взаимодействия	Сетевой стек ISO/OSI и TCP/IP. Сокеты. Маршрутизация. Работа с DNS.
8	Сетевое взаимодействие	Протокол HTTP. REST и RESTful API. Аутентификация и авторизация. Отладка API.
9	Контейнеризация и Docker	Виртуализация и контейнеризация. Docker, podman. Образы и контейнеры Docker и тома. Слои и их виды. Работа с Docker Registry. Продвинутые команды работы с контейнерами и образами
10	Непрерывная интеграция/Непрерывная доставка	CI: основные принципы - сборка и тестирование проектов. Артефакты сборки проектов. CD: доставка проектов - виды окружений, развертывание приложений по SSH.
11	Принципы написания программного кода	ИИ-разработка: провайдеры, безопасность, аудит. Понятие skills и agents.

4. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый слушатель в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Слушателям обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Менцер Филлиппо, Фортунато Санто, Дэвис Клейтон - Наука о сетях: вводный курс. ДМК Пресс, 2022 г., 338 с. https://znanium.ru/catalog/document?id=446466.
2. Гадзурас Эммануил - Docker Compose для разработчика - ДМК Пресс, 2023 г., 222 с. https://znanium.ru/catalog/document?id=445334.

Дополнительная литература:

1. Смарт Джон Фергюсон, Молак Ян - BDD в действии - ДМК Пресс, 2023 г., 554 с.https://znanium.ru/catalog/document?id=462885.

5. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и

обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование"	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		

Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

6. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Основы промышленной разработки» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, домашние задания, контрольная работа, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удается, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где слушатели активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в

обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал, использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Контрольная работа – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы – получить специальные знания по одной или нескольким темам дисциплины и продемонстрировать навыки их практического применения.

Самостоятельная работа – работа слушателей, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы слушатели взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи слушателя включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме ***зачета с оценкой***.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

Десятибалльная оценка	**Пятибалльная оценка**	**Оценка за зачет**	**Общая характеристика результата обучения по дисциплине (модулю)**
10	Отлично	Зачтено	Слушатель полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			задачи. Слушатель хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
7	Хорошо	Зачтено	Слушатель обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Слушатель хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Слушатель обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Слушатель способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Слушатель не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Основы промышленной разработки» оценивается следующим образом:

Домашние задания	50%	10	Набор задач по темам недели
Контрольная работа	25%	1	Письменная работа с набором задач, которые нужно решить за ограниченное время
Зачет с оценкой	25%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

Формула расчёта итоговой оценки по дисциплине (модулю) «Основы промышленной разработки»: «0,5 × среднее за домашние задания + 0,25 × контрольная работа + 0,25 × зачет с оценкой».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание 1. Работа с Git и командной строкой Linux

1. Установите Linux (или используйте WSL) и выполните базовую настройку.
2. Создайте локальный репозиторий с помощью git init.
3. Добавьте файл README.md, сделайте коммит с осмысленным сообщением.
4. Создайте ветку dev, внесите изменения в файл, закоммитите.
5. Переключитесь на основную ветку, слейте изменения из dev с использованием merge.
6. Настройте удалённый репозиторий на GitHub, отправьте изменения.
7. Оформите отчёт: скриншоты терминала с ключевыми командами и результатами.

Домашнее задание 2. Сборка и организация Python-проекта

1. Создайте структуру проекта: src/, tests/, pyproject.toml.
2. Установите poetry, инициализируйте проект, добавьте зависимости (requests, pytest).
3. Напишите простую функцию в src/utils.py (например, fetch_data(url)).
4. Создайте тест в папке tests/.
5. Соберите проект с помощью poetry build.
6. Оформите инструкцию по установке и запуску в README.md.

Домашнее задание 3. Работа с Docker и контейнеризацией

1. Напишите Dockerfile для Python-приложения (например, Flask-сервер, возвращающий "Hello, World!").
2. Соберите образ с тегом myapp:latest.
3. Запустите контейнер, пробросив порт 5000.
4. Создайте volume и подключите его к контейнеру для сохранения данных.
5. Опубликуйте образ в Docker Hub (или локальном registry).
6. Оформите пошаговую инструкцию в виде markdown-файла.

Примерные задания для контрольной работы

Задание 1. Работа с SSH и безопасным доступом к удалённым системам
Вы работаете в команде, где необходимо настроить безопасный доступ к удалённому серверу для нескольких разработчиков. Сервер работает под управлением Linux.

Требуется выполнить:

1. Описать пошагово, как сгенерировать SSH-ключ с помощью ssh-keygen (без пароля для простоты).
2. Объяснить, как добавить публичный ключ на удалённый сервер (через ~/.ssh/authorized_keys).
3. Написать команду для подключения к серверу по SSH: user@192.168.1.100.
4. Описать, как скопировать файл app.tar.gz с локальной машины на сервер в папку /home/user/deploy/ с помощью scp.
5. Указать, как проверить, что SSH-демон запущен на сервере, и как открыть порт 22 в ufw (файрвол).
6. Объяснить, почему использование SSH-ключей безопаснее, чем логин с паролем.

Задание 2. Анализ и оптимизация Dockerfile
Условие:
Дан Dockerfile для Python-веб-приложения:
dockerfile
```
FROM python:3.10
WORKDIR /app
COPY . .
RUN pip install -r requirements.txt
EXPOSE 8000
CMD ["gunicorn", "app:app", "--bind", "0.0.0.0:8000"]
```
Требуется выполнить:

1. Описать назначение каждой инструкции Dockerfile.
2. Указать, какие слои будут созданы и как они кэшируются.
3. Найти две проблемы в этом Dockerfile (например, отсутствие.dockerignore, нестабильный порядок копирования, отсутствие многоступенчатой сборки).
4. Переписать Dockerfile с учётом оптимизации:
    - Добавить.dockerignore,
    - Сначала копировать requirements.txt, устанавливать зависимости, потом — остальные файлы,
    - Использовать многоступенчатую сборку (на выбор: для уменьшения размера образа).
5. Объяснить, зачем нужен EXPOSE и обязательно ли пробрасывать порт при запуске.

Задание 3. Работа с REST API и отладка HTTP-запросов
Условие:
Вам необходимо интегрироваться с внешним REST API сервиса управления задачами. API требует аутентификации по токену и поддерживает следующие методы:

- GET /api/tasks — получить список задач

- POST /api/tasks — создать новую задачу
- PATCH /api/tasks/{id} — обновить задачу
- Authorization: Bearer <token>

Требуется выполнить:

1. Написать команду curl, которая отправляет POST-запрос для создания задачи с JSON-телом.
2. Указать, как с помощью curl сохранить ответ в файл response.json.
3. Написать команду curl, которая получает список задач и выводит HTTP-статус ответа.
4. Объяснить, чем отличается PUT от PATCH в REST.
5. Описать, как проверить, доступен ли API (например, с помощью curl -I или ping).
6. Указать, где в браузере или в терминале можно увидеть заголовки ответа от сервера.