

УТВЕРЖДАЮ

Индивидуальный предприниматель
Донской Александр Игоревич
Введено в действие с 27 июня 2025 года

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ

«DevOps Фабрика. Основная программа»
(уровень для продвинутых)

Направленность программы - техническая

Возраст обучающихся - от 18 лет и старше

Срок освоения образовательной программы: 6 месяцев

Общее количество часов освоения образовательной программы - 390 ак.ч.

Иркутская область, г. Иркутск
2025 г.

**Паспорт дополнительной общеобразовательной общеразвивающей программы
технической направленности «DevOps Фабрика. Основная программа»**

Название ДООП	«DevOps Фабрика. Основная программа»
Сведения об авторе/правообладателе	Индивидуальный предприниматель Донской Александр Игоревич
Нормативно-правовая база, использованная при подготовке ДООП	Дополнительная общеразвивающая программа дополнительного образования детей и взрослых (далее - образовательная программа; программа) разработана на основе: 1. Федеральный закон от 29.12.2012г. №273-ФЗ «Об образовании в Российской Федерации»; 2. Постановление Правительства Российской Федерации от 18.09.2020№ 1490 "О лицензировании образовательной деятельности"; 3. Приказ Министерства Просвещения России от 27.07.2022 № 629 “Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам”
Материально-техническая база	Освоение программы осуществляется удаленно с использованием функционала личного кабинета обучающегося, расположенного на информационном ресурсе в сети Интернет по ссылке https://factory.getcourse.ru , правом использования информационного ресурса обладает Донской Александр Игоревич на основании договора с ООО «Система Геткурс». Коммуникация с преподавателем программы осуществляются с помощью электронной почты и других средств связи, указанных в настоящей программе.
Год разработки	2025
Структура программы	Программа имеет модульную структуру и состоит из 26 тематических разделов, последовательно формирующих ключевые компетенции в области DevOps. Каждый модуль включает видеоматериалы, обязательные и дополнительные практические задания, а также сопровождается индивидуальной обратной связью от преподавателя по каждому выполненному заданию.
Направленность	техническая
Возраст учащихся	от 18 лет и старше
Срок реализации (освоения)	6 месяцев
Общее количество часов освоения	390 академических часов
Актуальность	Программа ориентирована на начинающих специалистов и охватывает ключевые навыки, востребованные в профессии DevOps-инженера. В неё входят работа с системами контроля версий (Git,

	GitHub, GitLab), автоматизация с помощью Bash, настройка сетей и защита серверов от атак. Эти компетенции соответствуют актуальным требованиям рынка и позволяют выполнять типовые задачи системного и сетевого администрирования.
Новизна	Программа включает современные инструменты DevOps-практики: создание CI/CD пайплайнов, настройка веб-сервера nginx, автоматизация с использованием Ansible, работа с базами данных (SQL и NoSQL) и средствами резервного копирования. Весь материал подается в практикоориентированном формате, что позволяет слушателю применять знания на практике уже в процессе обучения.
Цель	Целью данной программы является углубленное обучение специалистов в области DevOps, включающее расширение знаний и навыков работы с системами контроля версий, сетевой инфраструктурой, защитой от сетевых угроз, нагрузочным тестированием, а также управление конфигурацией и контейнеризацией приложений. Курс направлен на подготовку участников к решению сложных задач в рамках современных IT-инфраструктур.
Ожидаемые результаты	По завершению курса слушатель будет знать основы DevOps-практик, включая работу с Git и Bash, настройку Linux-серверов, веб-сервера nginx и межсетевых экранов, автоматизацию с помощью Ansible, работу с базами данных, контейнерами Docker и CI/CD-инструментами. Он сможет применять полученные знания на практике, участвовать в командной разработке и выполнять типовые задачи DevOps-инженера.
Форма занятий	Заочная форма обучения с использованием дистанционных технологий.
Форма текущего и итогового контроля	Каждую неделю выдается блок заданий, которые являются стоп-уроками, это означает, что для дальнейшего прохождения программы необходимо выполнить обязательный блок заданий. Помимо обязательного блока заданий есть факультативная часть, которая не обязательна к выполнению, но желательна для более полного освоения программы. Все задания проверяются вручную преподавателем, по ним дается обратная связь и комментарии, что нужно исправить.

СОДЕРЖАНИЕ

1. Пояснительная записка
2. Учебный план программы. Содержание рабочей программы
3. Организационно-педагогические условия реализации образовательной программы
4. Учебно-методическое обеспечение

1. Пояснительная записка

Дополнительная общеразвивающая образовательная программа предназначена для обучающихся, проявляющих повышенный интерес к информационным технологиям. Программа имеет практическую направленность с видеоматериалами (продолжительность видеоматериалов от 1 минуты до 15 минут). Сам процесс обучения происходит во время выполнения практических задач – даются подробные комментарии по коду (по его исправлению и улучшению).

1.1 Нормативные правовые основания разработки программы

Программа дополнительного образования Индивидуального предпринимателя Донского Александра Игоревича разработана на основе следующих правовых актов:

- Федеральный закон РФ от 29.12.2012 г. № 273-ФЗ “Об образовании в Российской Федерации”;
- Постановление Правительства Российской Федерации от 18.09.2020 № 1490 “О лицензировании образовательной деятельности”;
- Распоряжение Правительства от 24.11.2020 №3081-р “Об утверждении Стратегии развития физической культуры и спорта в РФ до 2030 года”;
- Приказ Министерства Просвещения России от 27.07.2022 № 29 “Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам”.

1.2 Цели программы

Целью данной программы является углубленное обучение специалистов в области DevOps, включающее расширение знаний и навыков работы с системами контроля версий, сетевой инфраструктурой, защитой от сетевых угроз, нагрузочным тестированием, а также управление конфигурацией и контейнеризацией приложений. Курс направлен на подготовку участников к решению сложных задач в рамках современных IT-инфраструктур.

1.3 Задачи программы:

- Освоение навыков работы с системами контроля версий, включая Git, GitLab и GitHub, разрешение конфликтов, использование продвинутых функций.
- Расширение знаний о работе с Bash, сетевая настройка, статическая маршрутизация, работа с межсетевыми экранами.
- Развитие умений по защите от сетевых атак с помощью fail2ban и проведению нагрузочного тестирования.
- Формирование знаний о типовых конфигурациях веб-сервера nginx для хостинга статических файлов и WSGI-приложений.
- Ознакомление с системой управления конфигурациями Ansible, её ролями, коллекциями, инвентарями, переменными и плейбуками.
- Изучение SQL и NoSQL баз данных, основ языка SQL, установки, создания пользователей, типовых операций и принципов организации резервного копирования.
- Освоение контейнеризации с использованием Docker, создание и оптимизация Dockerfile, управление ресурсами, multi-stage сборки и регистрация образов.
- Закрепление навыков докеризации приложений, написанных на разных языках программирования, с учетом специфики установки зависимостей, конфигурирования приложений и подключения к базам данных и другим сервисам.

- Создание и настройка CI/CD пайплайнов с использованием GitLab CI, GitHub Actions и Jenkins, включая сборку изображений, деплой контейнеров и проверку pull request'ов.
- Изучение observability – мониторинга и анализа логов с применением Zabbix, Grafana, Prometheus, Alertmanager, Telegraf, Elasticsearch, Kibana и Logstash/Fluentd.

1.4 Актуальность и новизна образовательной программы

Программа направлена на формирование у слушателей глубоких знаний и практических навыков в области DevOps, начиная с работы с системами контроля версий и заканчивая созданием и настройкой CI/CD пайплайнов. Особое внимание уделяется освоению продвинутых возможностей Git, разрешению конфликтов, использованию таких инструментов, как GitLab и GitHub, а также применению функций cherry-pick и revert. Программа охватывает расширенное изучение Bash, включая работу с текстовыми файлами и регулярными выражениями, а также настройку сетевой инфраструктуры, включая статическую маршрутизацию и работу с межсетевыми экранами iptables и ufw. Важным элементом является защита от сетевых атак с использованием fail2ban и проведение нагрузочного тестирования.

Далее программа фокусируется на изучении типовых конфигураций веб-сервера nginx для хостинга статических файлов и WSGI-приложений, а также на системе управления конфигурациями Ansible. Слушатели узнают о создании ролей, коллекций, управлении инвентарем, использовании переменных и плейбуков, а также о возможностях шифрования данных с помощью Ansible Vault. Программа также затрагивает работу с различными базами данных, включая SQL и NoSQL, установку, создание пользователей, выполнение типовых операций и организацию резервного копирования.

В случае успешного прохождения курса обучающийся получит четкое понимание работы DevOps'а, инструментов, основных задач, опыт и умения, которых достаточно для того, чтобы решать типовые задачи в профессии. По итогам обучения, с полученными навыками слушатель сможет участвовать в коммерческих проектах, и проходить собеседования по устройству на работу DevOps-инженером.

1.5 Категория слушателей:

Начинающие программисты, студенты технических специальностей, люди без опыта в IT, которые планируют начать свою карьеру в DevOps, и специалисты, которым нужны навыки в смежных профессиях.

1.6 Возраст обучающихся, ограничения

Программа предназначена для взрослых от 18 лет и старше.

1.7 Ожидаемые результаты:

В результате освоения программы обучающийся должен приобрести следующие знания, умения и навыки:

знать:

- основные принципы работы систем контроля версий Git, GitHub и GitLab, включая разрешение конфликтов и использование дополнительных функций (cherry-pick, revert и др.);
- структуру и синтаксис Bash, работу с текстовыми файлами и регулярными выражениями;
- основы сетевой настройки и статической маршрутизации в Linux;
- принципы работы межсетевых экранов (iptables, ufw) и систем защиты от атак (fail2ban);

- типовые конфигурации веб-сервера nginx для различных задач;
- архитектуру и возможности системы управления конфигурациями Ansible;
- различия между реляционными (SQL) и нереляционными (NoSQL) базами данных, принципы резервного копирования;
- основы контейнеризации и принципы работы Docker;
- элементы CI/CD-процессов и назначение инструментов GitLab CI, GitHub Actions, Jenkins;
- назначение и возможности систем мониторинга и анализа логов (Zabbix, Prometheus, Grafana и др.).

уметь:

- применять Git для управления версиями, ветвления и слияния кода в командной разработке;
- использовать Bash для написания скриптов и автоматизации задач;
- настраивать сетевые интерфейсы и маршруты, управлять правилами файрвола;
- конфигурировать nginx для обслуживания веб-приложений;
- создавать и применять роли, инвентари и переменные в Ansible;
- выполнять установку, настройку и базовые операции с базами данных;
- создавать и оптимизировать Dockerfile, собирать и запускать контейнеры;
- реализовывать CI/CD пайплайны с автоматической сборкой и деплоем приложений;
- организовывать мониторинг и сбор логов с помощью современных инструментов.

обладать навыками:

- интеграции Git в процессы DevOps и командной разработки;
- анализа и устранения сетевых проблем и угроз;
- автоматизации настройки серверов с использованием Ansible и Bash;
- работы с контейнеризированной инфраструктурой;
- обеспечения отказоустойчивости и наблюдаемости сервисов.

1.8 Форма обучения

Заочная форма обучения с применением исключительно электронного обучения с использованием дистанционных образовательных технологий. Обучающийся осуществляет обучение в индивидуальном порядке в соответствии с последовательно представленной информацией в программе образования.

1.9 Срок реализации программы

Срок реализации (освоения) программы - 6 месяцев (1 тема на 15 часов раз в неделю).

Общее количество часов освоения программы - 390 ак.ч.

1.10 Планируемые результаты обучения

Планируемые результаты обучения представлены в виде общих для программы навыков и умений, которые осваиваются в результате обучения по учебным курсам, входящим в учебный план программы.

2. Учебный план. Содержание рабочей программы

Список тем, которые будут изучены на курсе:

1. Расширенная работа с Git, Gitlab, Github, разрешение конфликтов, cherry-pick, revert. Расширенная работа с Bash.
2. Сетевые настройки, статическая маршрутизация, работа с межсетевыми экранами iptables и ufw, NAT. Защиты от сетевых атак с помощью fail2ban. Нагрузочное тестирование.
3. Типовые конфигурации веб-сервера nginx для хостинга статических файлов и для wsgi приложений.
4. Система управления конфигурациями ansible. Роли, коллекции, инвентори, переменные хоста и групповые переменные, плейбуки, ansible vault.
5. SQL и noSQL базы данных. Основы языка SQL. MySQL, PostgreSQL, MongoDB, Redis, Memcached, Rabbitmq. Установка, создание пользователей, типовые операции. Принципы организации резервного копирования.
6. Контейнеризация Docker, образы, контейнеры, docker-compose, подготовка и оптимизация Dockerfile, лимитирование ресурсов. Multi-stage билды. Docker registry.
7. Докеризация приложений, написанных на Python, GO, Node.js, установка зависимостей, конфигурация приложений, конфигурация подключения к базам данных и другим сервисам. Подготовка образов для их деплоя в тестовые и продакшн окружения.
8. CI/CD пайплайны. Gitlab CI, Github actions, Jenkins. Настройка типовых CI/CD процессов, сборка image, деплой контейнеров. проверки Pull Request'ов.
9. Observability. Мониторинг Zabbix, Grafana Prometheus, Alertmanager, Telegraf. Доставка и анализ логов в elasticsearch, kibana, logstash/fluentd.

2.1 Учебный план

N п/п	Название раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1	Onboarding	15	1	14	Практическое задание
2	Password manager и rsync	15	-	15	Практическое задание
3	Branching model	15	-	15	Практическое задание
4	Bash скриптинг и nginx	15	-	15	Практическое задание
5	Сети: фаерволы и статическая маршрутизация	15	-	15	Практическое задание
6	Nginx и NAT	15	-	15	Практическое задание
7	Регистрация сервиса, Mysql	15	-	15	Практическое задание
8	Postgres	15	-	15	Практическое задание
9	Ansible и настройка redis	15	-	15	Практическое задание
10	Rabbitmq	15	-	15	Практическое задание
11	Mongodb и Memcached	15	-	15	Практическое задание
12	Docker	15	-	15	Практическое задание

13	Контейнеризация GO приложения	15	-	15	Практическое задание
14	Контейнеризация Python приложения, docker registry	15	-	15	Практическое задание
15	Контейнеризация Nodejs приложения, harbor docker registry	15	-	15	Практическое задание
16	CI/CD с Github actions	15	-	15	Практическое задание
17	PR проверки	15	-	15	Практическое задание
18	Gitlab CI и миграции схемы БД	15	-	15	Практическое задание
19	Gitlab реро и динамические тестовые окружения	15	-	15	Практическое задание
20	Первый этап итого проекта: CI/CD для деплоя инфраструктуры	15	-	15	Итоговый проект
21	Второй этап. CI/CD приложения и упражнения с git	15	-	15	Итоговый проект
22	Третий этап Observability: Zabbix	15	-	15	Итоговый проект
23	Третий этап Observability: Prometheus	15	-	15	Итоговый проект
24	Четвертый этап: Logging	15	-	15	Итоговый проект
25	Load testing	15	-	15	Итоговый проект
26	Architecture diagram	15	-	15	Итоговый проект
	ИТОГО	390	1	389	

2.2 Содержание рабочей программы

Тема 1: Onboarding	В рамках данной темы учащиеся узнают: Основы работы с SSH-ключами, генерация ключей разных типов, приватная и публичная части, конвертация форматов. Основы работы с GitHub, двухфакторная аутентификация, управление SSH-ключами, работа с ветками и pull request.	Практика. Основные концепты GIT, авторизация в удаленных репозиториях. Создание запросов на изменения и работа с ветвлением.
Тема 2	В рамках данной темы учащиеся узнают: Выбор и настройка менеджера паролей. Написание bash-скриптов для управления пользователями.	Практика. Создание пользователей в операционных системах Linux. Работа с файлами с использованием утилит rsync. Практики безопасного хранения паролей.

	Работа с rsync через SSH. Использование ssh-agent, управление дисковым пространством, создание кастомного shell.	
Тема 3: Branching model	В рамках данной темы учащиеся узнают: Работа с ветками в Git, разрешение конфликтов, squash merge. Мониторинг системных ресурсов. Использование screen и tmux.	Практика. Разрешение конфликтов в GIT, оценка степени загрузки серверов под управлением Linux.
Тема 4: Bash скриптинг и nginx	В рамках данной темы учащиеся узнают: Основы скриптового языка bash. Установка и настройка nginx. Работа с basic auth и проксированием.	Практика. Написание скриптов автоматизации на bash. Установка и первоначальная настройка вебсервера nginx.
Тема 5: Сети: фаерволы и статическая маршрутизация	В рамках данной темы учащиеся узнают: Базовая настройка файрвола с iptables, ufw и nftables. Настройка статической маршрутизации.	Практика. Настройки межсетевых экранов (iptables, ufw, nftables), маршрутизация и диагностика сетевых соединений в виртуальной среде.
Тема 6: Nginx и NAT	В рамках данной темы учащиеся узнают: Принципы настройки защищенного веб-сервера, основы сетевого взаимодействия и маршрутизации, методы анализа сетевого трафика и мониторинга доступности сервисов.	Практика. Настройка защищенного веб-сервера, базовые принципы маршрутизации и диагностики сети, мониторинг доступности сервисов.
Тема 7: Регистрация сервиса, Mysql	В рамках данной темы учащиеся узнают: Как автоматизировать выполнение фоновых задач, управлять системными сервисами и настраивать доступ к данным с учетом требований безопасности и надежности.	Практика. Автоматизация фоновых задач, введение в управление сервисами и базами данных, основы администрирования доступа и резервного копирования.
Тема 8: Postgres	В рамках данной темы учащиеся узнают: Подходы к автоматизации системных процессов, планированию задач и настройке баз данных с контролем доступа и соблюдением требований безопасности.	Практика. Автоматизация системных процессов, настройка реляционной БД с управлением доступом и планирование фоновых задач.

Тема 9: Ansible и настройка redis	В рамках данной темы учащиеся узнают: Базовые принципы конфигурационного управления и автоматизации настройки сервисов в инфраструктуре.	Практика. Основы конфигурационного управления: организация Ansible-проектов, автоматизация настройки сервисов и пользователей.
Тема 10: RabbitMQ	В рамках данной темы учащиеся узнают: Как автоматизировать настройку распределённых сервисов, управлять правами доступа и обеспечивать отказоустойчивость в кластерной инфраструктуре.	Практика. Автоматизация развертывания и конфигурации кластерных сервисов с использованием Ansible, настройка отказоустойчивости.
Тема 11: Mongodb и Memcache d	В рамках данной темы учащиеся узнают: Подходы к работе с системами хранения данных и кэширования, настройке доступа, резервному копированию и обеспечению отказоустойчивости сервисов.	Практика. Настройка NoSQL-хранилищ и кэш-сервисов, автоматизация установки, резервное копирование и проверка отказоустойчивости.
Тема 12: Docker	В рамках данной темы учащиеся узнают: Основы контейнеризации, развертывание приложений в контейнерах, управление окружением и базовые практики обеспечения безопасности сервисов.	Практика. Развертывание и конфигурация сервисов в контейнерах, настройка окружения и базовые подходы к обеспечению безопасности.
Тема 13: Контейнер изация GO приложени я	В рамках данной темы учащиеся узнают: Принципы сборки и упаковки приложений в контейнеры, работу с системой контроля версий и оформление проектной документации.	Практика. Основы сборки и автоматизации контейнеризированных приложений, подготовка окружения и проектной документации.
Тема 14: Контейнер изация Python приложени я, docker registry	В рамках данной темы учащиеся узнают: Как развертывать и управлять многокомпонентными приложениями в контейнерах, настраивать окружение и организовывать хранение образов.	Практика. Основы управления многоконтейнерными приложениями, публикация и хранение образов, настройка окружения для сервисов.
Тема 15: Контейнер изация Nodejs приложени я, harbor docker registry	В рамках данной темы учащиеся узнают: Подходы к контейнеризации веб-приложений, управлению процессами в контейнерах и обеспечению безопасности при хранении образов.	Практика. Контейнеризация приложений и организация безопасного хранения образов с управлением доступом.
Тема 16: CI/CD с	В рамках данной темы учащиеся узнают: Основы	Практика. Настройка процессов непрерывной интеграции и доставки,

Github actions	автоматизации процессов разработки и развертывания, настройку инструментов CI/CD и управление инфраструктурой.	автоматизация сборки и развертывания сервисов.
Тема 17: PR проверки	В рамках данной темы учащиеся узнают: Как интегрировать автоматизированные проверки качества и тестирования кода в процессы разработки и доставки.	Практика. Внедрение автоматизации проверки и тестирования кода в процессы разработки и деплоя.
Тема 18: Gitlab CI и миграции схемы БД	В рамках данной темы учащиеся узнают: Принципы автоматизации процессов миграции баз данных и их тестирования в рамках CI/CD на собственных инфраструктурных решениях.	Практика. Организация и автоматизация процессов CI/CD с использованием self-hosted решений.
Тема 19: Gitlab hero и динамические тестовые окружения	В рамках данной темы учащиеся узнают: Методы автоматизации и управления процессами разработки, включая настройку защиты кода, создание динамических тестовых окружений и интеграцию с сетевой инфраструктурой.	Практика. Автоматизация рабочих процессов и настройка инфраструктуры для эффективной поддержки разработки и деплоя.
Тема 20: Первый этап итогового проекта: CI/CD для деплоя инфраструктуры	В рамках данной темы учащиеся узнают: Как применять инструменты CI/CD и контейнерные технологии для автоматизации развертывания инфраструктуры в рамках итогового проекта.	Практика. Автоматизация развертывания и мониторинга ключевых сервисов с обеспечением стабильной и бесперебойной работы.
Тема 21: Второй этап. CI/CD приложения и упражнения с git	В рамках данной темы учащиеся узнают: Как применять навыки автоматизации CI/CD и управления версиями для комплексного развертывания итогового проекта.	Практика. Создание и автоматизация процессов непрерывного развертывания веб-приложений и управления версиями.
Тема 22: Третий этап Observability: Zabbix	В рамках данной темы учащиеся узнают: Как интегрировать мониторинг и систему оповещений в итоговый проект для отслеживания состояния сервисов и обеспечения их стабильности.	Практика. Настройка мониторинга и оповещений для приложений и сервисов с использованием систем сбора метрик и уведомлений.

Тема 23: Третий этап Observability: Prometheus	В рамках данной темы учащиеся узнают: Как внедрить мониторинг с использованием Prometheus, организовать сбор метрик, визуализацию и оповещения в рамках итогового проекта.	Практика. Организация сбора метрик и мониторинга систем с настройкой оповещений и визуализации данных.
Тема 24: Четвертый этап Logging	В рамках данной темы учащиеся узнают: Как реализовать централизованный сбор, хранение и анализ логов для приложений и инфраструктуры в рамках итогового проекта.	Практика. Настройка централизованного сбора и анализа логов с применением современных инструментов и оповещений.
Тема 25: Load testing	В рамках данной темы учащиеся узнают: Как проводить нагрузочное тестирование приложений, интерпретировать результаты и интегрировать его в процессы наблюдаемости и CI/CD.	Практика. Оценка производительности приложений и автоматизация тестирования с интеграцией оповещений.
Тема 26: Architecture diagram	В рамках данной темы учащиеся узнают: Как описывать архитектуру проекта, учитывать требования отказоустойчивости и безопасности, а также автоматизировать реагирование на инциденты.	Практика. Проектирование устойчивых систем, обеспечение безопасности и автоматизация управления инцидентами.

3. Организационно-педагогические условия реализации образовательной программы

3.1 Форма обучения:

Очное с применением исключительно электронного обучения, дистанционных образовательных технологий.

Календарный учебный график

- **Старт обучения:** Обучение начинается индивидуально для каждого слушателя сразу после записи на курс.
- **Окончание обучения:** По мере прохождения учеником основной программы, согласно индивидуальному расписанию и определяется по согласованию со слушателем программы.

3.2 Форма организации образовательной деятельности обучающихся

Уроки в виде видеоматериалов и описания практических заданий находятся в личном кабинете обучающегося на информационном ресурсе (сайте) в <https://factory.getcourse.ru>. Все уроки сети Интернет по ссылке дополнительной образовательной программы открываются у обучающегося постепенно, 1 раздел в неделю. Обучающиеся могут проходить уроки в любое удобное для них время. Уроки, входящие в состав дополнительной образовательной программы, расположены в системной, последовательной форме, и состоят из практического материала по программе. Процесс обучения должен быть построен следующим образом: 1 (один) день обучения чередуется с 1 (одним) или 2 (двумя) днями отдыха по выбору обучающегося в зависимости от его физических возможностей. Обучающийся осуществляет обучение по программе дополнительного образования руководствуясь рекомендациями уроков программы. Уроки построены в соответствии с принципами обучения развивающего характера: доступности, наглядности, целенаправленности, индивидуальности, результативности, а также другими принципами.

В рамках прохождения образовательной программы проводятся консультации на платформе GetCourse: обучающиеся имеют возможность задать интересующие их вопросы по материалу, изученному на уроках, возникающим сложностям, а также условиям достижения наилучших результатов в сети в комментариях к уроку, а также в Телеграм в чате “DevOps Factory: Клуб при фабрике”. Вступить в телеграм-чат можно только по ссылке-приглашению, т.к. чат предназначен исключительно для обучающихся на курсе и участников клуба. Консультационная и информационная поддержка слушателей осуществляется службой технической поддержки на сайте <https://factory.getcourse.ru>.

3.3 Материально-технические условия реализации программы

Обучающийся осваивает образовательную программу полностью удаленно с использованием функционала информационного ресурса в сети Интернет, расположенного по ссылке <https://factory.getcourse.ru>, правом использования ресурса обладает Донской Александр Игоревич на основании договора от 07.05.2025 №777 между Обществом с ограниченной ответственностью «Система Геткурс» и индивидуальным предпринимателем Донским Александром Игоревичем (действует 1 год).

Все коммуникации с преподавателем осуществляются с помощью платформы Getcourse и телеграм-чата. Каждому обучающемуся программы дополнительного образования предоставляется доступ путем передачи на электронную почту логина и пароля к личному кабинету на информационном ресурсе в сети Интернет.

Информационное обеспечение

- Видеоматериалы;
- раздаточные материалы;
- литература;
- бонусные уроки;
- дополнительные материалы.

Программа реализуется исключительно с применением электронного обучения и дистанционных образовательных технологий «GetCourse» на платформе по адресу: <https://factory.getcourse.ru>.

Необходимое техническое обеспечение для прохождения обучения

Для успешного обучения с использованием электронного обучения обучающимся рекомендуется соблюдать определенные требования к программному обеспечению персонального компьютера:

- подключение к сети Интернет со скоростью минимально 2 Мбит/с, доступ к сети по протоколам HTTP;
- на компьютере также должен быть установлен комплект соответствующего программного обеспечения (Windows 7 / macOS 10.10 / Linux или выше)

Также необходимо наличие динамиков (наушников).

Для просмотра электронных образовательных ресурсов необходимо входить в личный кабинет на портале через:

- Компьютер, необходима современная версия браузера: Google Chrome, Mozilla Firefox, Microsoft Edge, Safari.
- Телефон, версия: iOS 9.x или выше (для iPad и iPhone); Android 4.4 или выше.

Используемая образовательная платформа адаптирована под мобильные устройства, однако для полноценного прохождения программы и выполнения практических заданий рекомендуется использование персонального компьютера или ноутбука.

4. Учебно-методическое обеспечение

1. *Федеральный закон от 29.12.2012 N 273-ФЗ "Об образовании в Российской Федерации"*.
2. *Приказ Министерства просвещения РФ от 22 марта 2021 г. No 115 "Об утверждении Порядка организации и осуществления образовательной деятельности по основным общеобразовательным программам - образовательным программам начального общего, основного общего и среднего общего образования"*.
3. ISBN 978-5-496-02303-0. Уильям Шоттс: *Командная строка Linux. Полное руководство*. 2-е изд. 2020. — 480 с.
4. ISBN 978-5-4461-3946-0. Брайан Уорд: *Внутреннее устройство Linux*. 3-е изд. 2025. — 384 с.
5. ISBN 978-5-4461-1514-3. Тронкон Пол, Олбинг Карл: *Bash и кибербезопасность: атака, защита и анализ из командной строки Linux*. 2022. — 288с.
6. ISBN 978-5-9775-6699-5. Фленов М.Е.: *«Linux глазами хакера»* 6-е изд. 2021. — 416 с.
7. ISBN 978-5-9775-1855-0. Джеймс Стронг, Валлери Лэнси: *Kubernetes и сети. Многоуровневый подход*. 2024. — 320 с.
8. ISBN 978-601-81034-1-4. Жюльен Пивотто, Брайан Бразил: *Запускаем Prometheus*. 2-е изд. 2023. — 392 с.
9. ISBN 978-6-01763-867-2. Бас Мейер, Лорин Хошштейн и Рене Мозер: *Запускаем Ansible*. 3-е изд. 2023. — 482 с.
10. ISBN 978-5-4461-1590-7. Брикман Евгений: *Terraform: инфраструктура на уровне кода*. 2-е изд. 2020. — 368 с.
11. ISBN 978-5-97060-772-5. Иан Милл, Эйдан Хобсон Сейерс: *Docker на практике*. 2-е изд. 2020. — 516 с.